.1

أجهزة الكمبيوتر: عملية البدء

مفاهيم أساسية

ما هو الكمبيوتر؟
- ☐ المعطيات والمعلومات
- ☐ معالجة المعطيات
- ☐ مفهوم البرنامج المخزّن

نظام الكمبيوتر
- ☐ مكوّنات النظام

كيف يعمل الكمبيوتر

ما هو الكمبيوتر؟

المعطيات والمعلومات

الكمبيوتر آلة وظيفتها قبول **المعطيات** ومعالجتها لتحويلها الى **معلومات**. والمعطيات هي حقائق أو ملاحظات بينما المعلومات هي المعاني التي ننسبها اليها.

لنطرح مثالاً لتوضيح ذلك: قضى عالم فلك العصور الوسطى تايكو براهي طيلة شبابه في مراقبة الكواكب وتدوين مواقعها. قام بجمع المعطيات، ففي ليلة من الليالي احتل المريخ موقعاً ما في السماء فأخذ يسجّل مجلّدات من هذه المعطيات، لكنه لم يكن متيقّناً أبداً ممّا تعنيه هذه المعطيات.

غير ان خلفه يوهانس كيبلر كان قد أدرك ان هناك نمطاً معيّناً، إذ وجد ان مدار المريخ يتخذ شكلاً اهليليجياً. وقد قضى معظم حياته يعالج المعطيات التي أوجدها براهي فقام بإجراء الحسابات المطوّلة وأعاد تنظيم ملاحظاته من أجل التثبّت من ذلك النمط. وفي نهاية المطاف نجح في نشر قوانينه المعروفة حول حركة الكواكب في عام 1621.

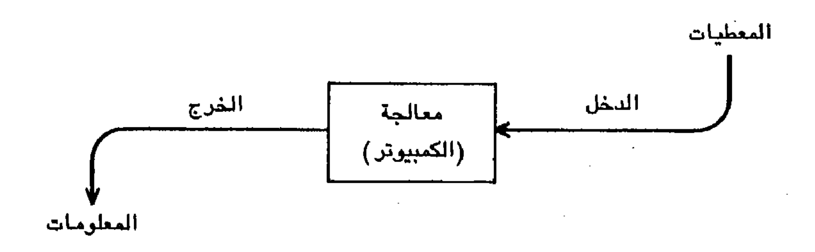
وتُعدّ قوانين كيبلر بمثابة معلومات، تمكّن بواسطتها أن يتفهّم حركة الكواكب ويتنبّأ بها. ولا يزال العلماء والمهندسون يعتمدون على قوانينه في تخطيط الرحلات الفضائية. إن للمعلومات معان.

ومن الواضح، ان قوانين كيبلر اشتقّت من المعطيات التي أوجدها براهي، لكن لا جدوى من المعطيات الخام اذا لم تتمّ معالجتها، فقبل أن يتمّ تنظيمها وإجراء الحسابات اللازمة عليها كانت المعطيات حقائق بلا هيكل أو معنى واضح. فمعرفة الموقع الدقيق للمريخ في اليوم الأول من شهر ابريل عام 1599 قد تعود على المرء بنقطة اضافية في مسابقة «تريفيال بيرسوت» (ملاحقة الوهم) لكن تلك الحقيقة ليست ذات فائدة تُذكَر، فبمعالجة المعطيات يمكن استخراج معانيها.

معالجة المعطيات

- **الكمبيوتر** هو آلة لمعالجة المعطيات. ويوضح الشكل 1.1 دفق المعطيات الى الآلة ويُسمّى **بالدخل**، وتُسمّى المعلومات المتدفّقة من الآلة **بالخرج**. ويعالج الكمبيوتر المعطيات. وقد أمضى يوهانس كيبلر عشرين سنة من حياته وهو يعالج المعطيات، أمّا اليوم فإن في وسع طالب جامعي يستخدم الكمبيوتر أن ينجز حسابات كيبلر كلها خلال ساعات قليلة.

الشكل 1.1 الكمبيوتر آلة تعالج المعطيات في صيغة معلومات. يستقبل الكمبيوتر معطيات الدخل ويجري معالجتها وإنتاج المعلومات كخرج.



ماذا نعني عندما نقول أن الكمبيوتر هو آلة تعالج المعطيات؟ ففي مجال التصنيع نعالج الحديد الخام لنصنع الفولاذ، ونعالج لب الخشب لصناعة الورق. إذاً فالمعالجة تتضمن إحداث تغيير وأن المواد الخام تجري إعادة تركيبها أو تشكيلها بطريقة ما. وفي كثير من الأحيان، تتضمن عملية المعالجة ترشيح المعطيات وتلخيصها وذلك لاستيعاب الأنماط الضمنيّة للمعطيات. أمّا كيف يقوم الكمبيوتر بمعالجة المعطيات وما الوظائف والعمليات التي يمكنه إنجازها فيمكن القول بشكل عام، ان في وسع أجهزة الكمبيوتر إجراء عمليات الجمع والطرح والضرب والقسمة والمقارنة والنسخ وكذلك طلبات الدخل والخرج. ولكن يمكن لآلات الجيب الحاسبة أداء العمليات نفسها. فما الذيّ يميّز الكمبيوتر عن غيره من الآلات الأخرى؟

مفهوم البرنامج المخزّن

لكي تقوم بجمع عددين مستخدماً الآلة الحاسبة فأنت:

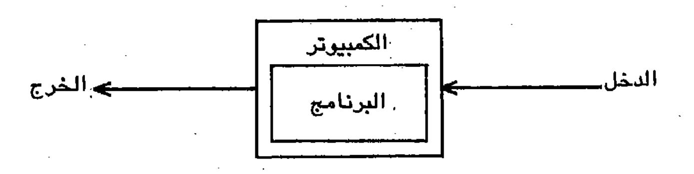
1. تدخل العدد الأول.
2. تكبس مفتاح علامة الجمع (+).
3. تدخل العدد الثاني.
4. تكبس مفتاح علامة التساوي (=).
5. تسجّل المجموع للرجوع اليه فيما بعد.

ان الآلة الحاسبة تعنى بالمجموع، لكن عليك التحكّم بتحديد المفتاح الذي تضبط عليه في الخطوة التالية. فإن الآلة الحاسبة تتطلّب تدخّلاً بشرياً في كل خطوة من الخطوات.

ويعالج الكمبيوتر المعطيات أوتوماتيكياً دون أي تدخّل بشري. ولكن أجهزة الكمبيوتر ليست ذكيّة، لأنها لا تعرف متى تجمع أو تطرح أو تقارن أو تجري عملية دخل. وإذا كان على الكمبيوتر أداء وظيفة ما دون تدخّل بشري مباشر، فلا بدّ من تزويده بمجموعة من التعليمات لتقوده خطوة خطوة طوال العملية. وهذه المجموعة من التعليمات تُسمى «**البرنامج**». ويخزّن البرنامج داخل الكمبيوتر ويصبح بذلك **برنامجاً مخزّناً** (الشكل 1.2). ويميّز البرنامج المخزّن الكمبيوتر عن الآلة الحاسبة ويتيح له إنجاز الوظائف دون أي تدخّل بشري. لندمج هذه الفكرة في التعريف التالي:

الكمبيوتر: آلة تعالج المعطيات لتصبح معلومات يحكمها برنامج مخزّن.

الشكل 1.2 يعالج الكمبيوتر المعطيات أوتوماتيكياً بتحكّم من برنامج مخزّن.

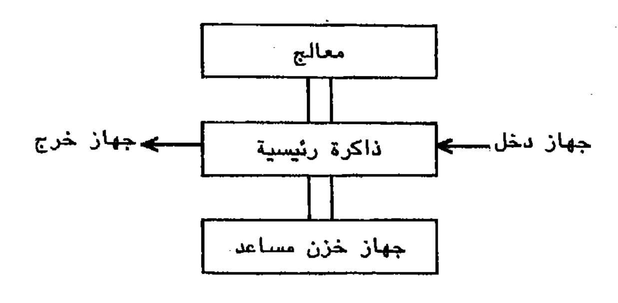


نظام الكمبيوتر

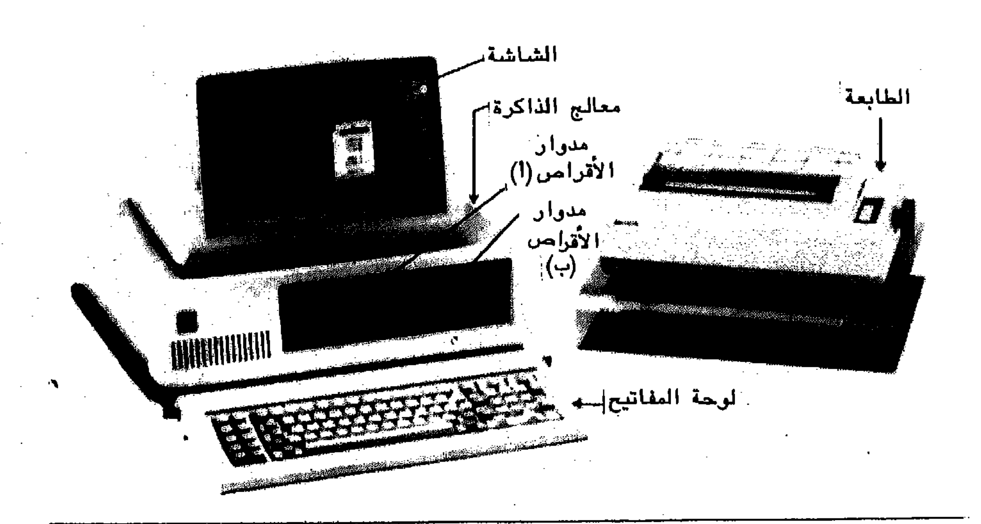
مكوّنات النظام

يتألف نظام الكمبيوتر من عدّة مكوّنات أساسية (الشكل 1.3). فإن جهاز الدخل يقدّم المعطيات، وهذه يجري تخزينها في **الذاكرة** التي تحتوي أيضاً على برنامج. يقوم **جهاز المعالجة** في الكمبيوتر بمعالجة المعطيات ضمن نطاق تحكّم ذلك البرنامج، ثم يخزّن النتائج ثانية في الذاكرة. وفي النهاية، تتدفّق النتائج من الكمبيوتر الى جهاز خرج. بالاضافة الى ذلك، فإن معظم أجهزة الكمبيوتر الحديثة تستخدم جهاز خزن مساعداً لزيادة سعة الذاكرة.

الشكل 1.3 يتألف نظام الكمبيوتر من أربعة مكوّنات أساسية: جهاز الدخل وجهاز الخرج والذاكرة الرئيسية وجهاز المعالجة. وغالباً ما يستخدم جهاز الخزن المساعد للزيادة في سعة الذاكرة.



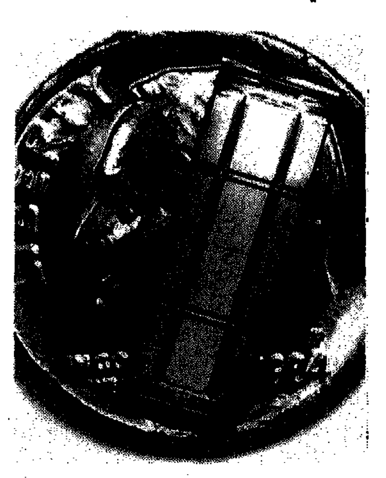
الشكل 1.4 نظام كمبيوتر نموذجي. تدخل المعطيات بواسطة لوحة المفاتيح يرسل الخرج الى الشاشة أو الطابعة. يوجد كل من المعالج والذاكرة الرئيسية داخل الصندوق المستطيل. مدوارا الأقراص يتيحان قدرة خزن مساعد.



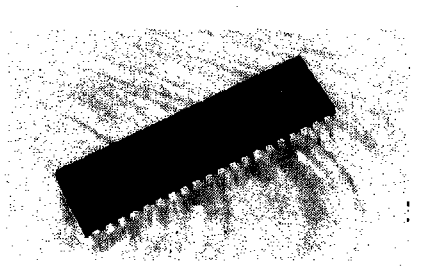
لنأخذ نظام الكمبيوتر المصوّر في الشكل 1.4 ففي مقدّمة الصورة تبدو لوحة المفاتيح التي تستخدم كجهاز دخل. وفوق لوحة المفاتيح يوجد جهاز الخرج وشاشة العرض. تُعتبر صور المعطيات المعروضة على الشاشة موقّتة ويمكننا الحصول على نسخة مطبوعة بارسال الخرج الى الطابعة. يوجد كل من المعالج والذاكرة الرئيسية داخل الخزانة (الصندوق المستطيل) قرب وسط الصورة. يساعد مدوار الأقراص على واجهة الصندوق المستطيل على زيادة سعة ذاكرة الكمبيوتر، وغالباً ما تدخل النظام برامج عن طريق أجهزة مثل أجهزة الخزن المساعد.

تُعدّ الرقيقة أو الجذاذة Chip (الشكل 1.5) اللبنة الأساسية لبناء جهاز كمبيوتر عصري. وهي دارة الكترونية متكاملة ومعقّدة التركيب يتمّ نمشها أو حفرها على طبقة مربّعة صغيرة مصنوعة من مادة السليكون لا يتعدّى حجمها حجم ظفر الاصبع. وبما أن الرقيقات المنفردة هشّة يصعب تناولها لذلك يجري تغليفها (الشكل 1.6) وتركيبها عادة على ألواح خاصة (الشكل 1.7). وقد يحتوي اللوح الأول المعالج واللوح الثاني الذاكرة الرئيسية واللوح الثالث الالكترونيات الخاصة بوصل جهاز دخل أو جهاز خرج بالنظام.

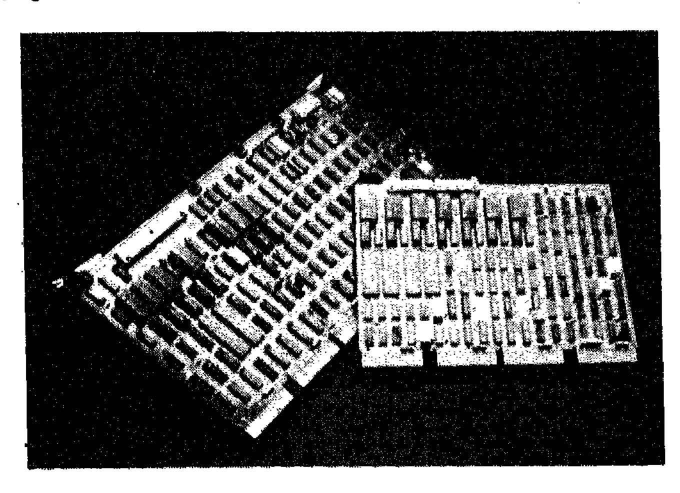
الشكل 1.5 الرقيقة أو الجذاذة هي اللبنة الأساسية لبناء كمبيوتر عصري.



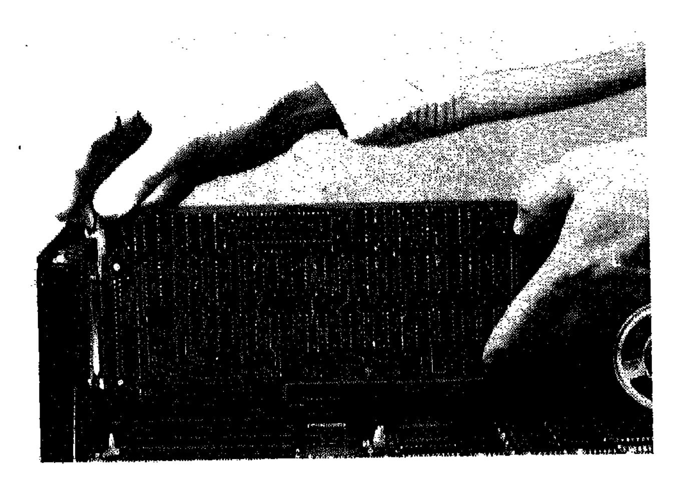
الشكل 1.6 الرقيقات هشّة يصعب تناولها، لذلك يجري عادة تغليفها وتركيبها على ألواح حاملة قبل استعمالها.



الشكل 1.7 تتألف مكوّنات الكمبيوتر الرئيسية من عدد من الرقيقات المركبة على لوح.



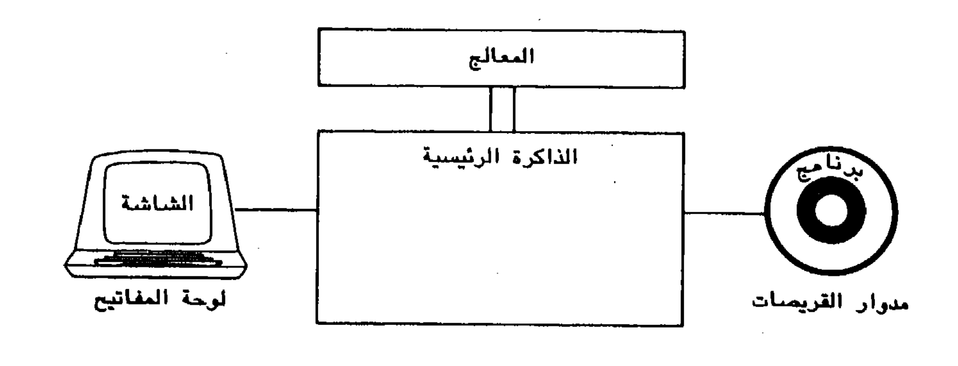
الشكل 1.8 يجري تجميع الكمبيوتر بزلق ألواح الرقيقات اللازمة في الخزانة الخاصة به.



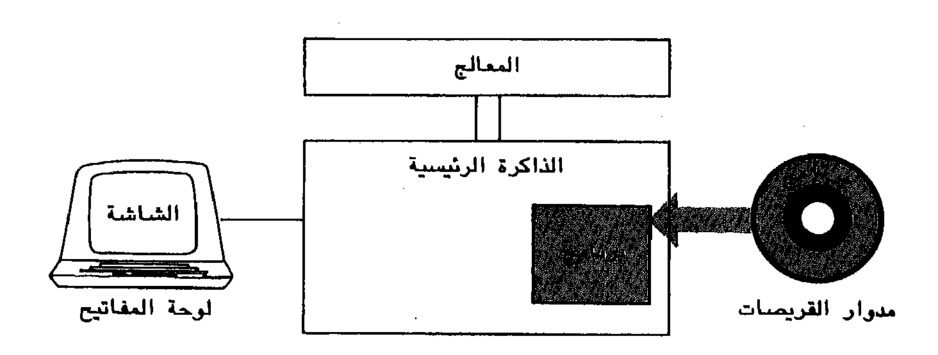
كيف يعمل الكمبيوتر

لنستعن بصورة النظام المبيّنة في الشكل 1.9 لتوضيح الطريقة النموذجية التي يعمل الكمبيوتر بها : يتمّ التحكّم بالكمبيوتر عن طريق البرنامج المخزّن، لذا فأول خطوة تُتخذ في استخدام الآلة هي نسخ البرنامج من القريصة المرنة في الذاكرة (الشكل 1.10أ). وهكذا يستطيع المعالج تنفيذ التعليمات. ويجري تخزين المعطيات المدخلة عن طريق لوحة المفاتيح في الذاكرة (الشكل 1.10ب). يقوم المعالج بمعالجة المعطيات وإعادة خزن النتائج في الذاكرة (الشكل 1.10ج). في النهاية تمثّل النتائج الخرج (الشكل 1.10د).

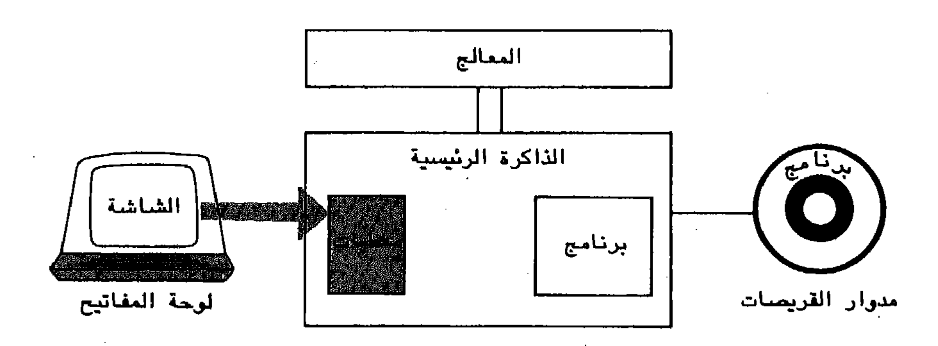
الشكل 1.9 سنستخدم الرسم التوضيحي التالي لنظام نموذجي في توضيح كيفية عمل الكمبيوتر.



الشكل 1.10 توضح هذه السلسلة من الرسوم كيفية عمل الكمبيوتر.

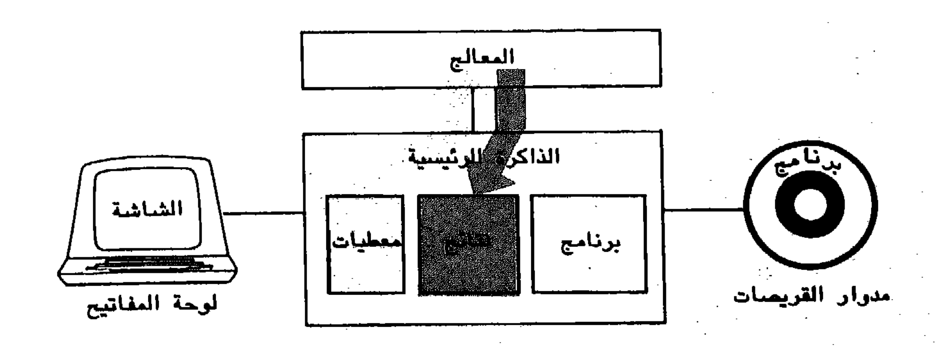


أ. يُنسَخ البرنامج عن القرص ويُخزَّن في الذاكرة.

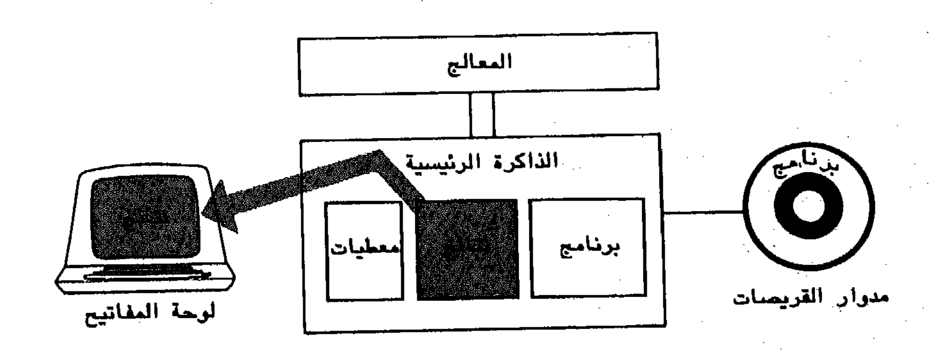


ب. تُقرأ المعطيات عن طريق لوحة المفاتيح وتُخزَّن في الذاكرة وذلك في نطاق التحكّم بواسطة البرنامج المُخزَّن.

الشكل 1.10



ج. يقوم المعالج بمعالجة المعطيات ويعيد خزن النتائج في الذاكرة.



د. في النهاية، تمثّل النتائج الخرج.

من السهل تغيير محتويات الذاكرة. لذلك، فعندما تتمّ معالجة فقرة واحدة من المعطيات، يمكن تنفيذ البرنامج ثانية، وذلك بقراءة ومعالجة معطيات جديدة وإحداث خرج جديد. وإذا أمكن تغيير المعطيات فلمَ لا يتمّ ذلك بالنسبة للبرنامج؟ لدى انتهاء برنامج ما، يحلّ محلّه برنامج آخر في الذاكرة مما يتيح للكمبيوتر معالجة معطيات تختلف اختلافاً كلياً عن سابقتها. ففي دقيقة معيّنة يمكنه إنتاج شيكات الرواتب المستنتجة من معطيات العمال بتحكّم من برنامج رواتب الموظفين. وفي الدقيقة التالية يمكنه تحضير الفواتير المستخرجة من بيانات الفوترة بتحكّم من برنامج الفوترة. وفي وقت لاحق يمكن لمجموعة البيانات نفسها بتحكّم من برنامج آخر، قراءة المعطيات الاحصائية وإحداث جدول أعمدة أو استشعار حركة المسلاة والتحكّم بموقع سفينة فضاء خيالية على الشاشة.

وأمّا مكوّنات الكمبيوتر المادية ـ المعالج والذاكرة وأجهزة الدخل وأجهزة الخرج ـ فهي تؤلّف **كيانه المادي** لأنه يمكنك مشاهدتها ولمسها وإدراك وجودها. أمّا البرامج والمعطيات فتختلف عن ذلك، لأنها توجد فقط في شكل نبضات الكترونية تُخزّن في الذاكرة. تُسمّى البرامج بشكل عام **بالكيان المنطقي**.

خطة الاستخدام

عند هذه النقطة، لا بدّ أن تكون قد اكتسبت معرفة أساسية بوظائف كل من المعالج والذاكرة الرئيسية وأجهزة الدخل والخرج وأجهزة الخزن المساعد، كما يجب أن تكون قد عرفت أن الكمبيوتر يعالج المعطيات في شكل معلومات بتحكّم من البرنامج. وسنناقش في الفصول المتعدّدة اللاحقة كل مكوّن من المكوّنات الرئيسية بشكل أكثر تفصيلاً بدءاً بالكمبيوتر نفسه. بعد إتمام معالجة المكوّنات سنتطرّق الى طريقة تجميعها بحيث تشكّل نظاماً كمبيوترياً كاملاً. ولمساعدتك في اختبار مدى استيعابك للمادة فقد أنهينا كل فصل من الفصول باختبار ذاتي قصير.

من خلال دراستك للمعالج أو الذاكرة أو أجهزة الدخل أو الكيان المنطقي، يجب أن لا تغيب عن ذهنك الحقيقة القائمة وهي أن الكمبيوتر هو مجموعة من المكوّنات التي يقوم كل منها بوظيفته الخاصة به. ولقد كان الغرض الأساسي من الفصل الأول هذا، تزويدك بلمحة عامة عن نظام الكمبيوتر الكامل. وكلّما توغّلت في الفصول التالية ستواجه معلومات فنية مفصّلة ومن السهل على المرء «أن يحفظ شيئاً وتغيب عنه أشياء». استعن بمادّة هذا الفصل كإطار عمل كلّما استغرقت في القراءة.

مصطلحات أساسية

ستجد في نهاية كل فصل من الفصول قائمة بالمصطلحات الرئيسية. ويجب عليك معرفة معانيها. وإذا لم تعرف فقد فاتك شيء وعليك أن تعيد قراءة المادة. بعد أن تكون قد عرفت كل مصطلح بنفسك، قارن أجوبتك بالمسرد.

- ☐ كمبيوتر
- ☐ معطيات
- ☐ كيان مادي
- ☐ معلومات
- ☐ دخل
- ☐ ذاكرة
- ☐ خرج
- ☐ معالج
- ☐ برنامج
- ☐ كيان منطقي (برامجيات)
- ☐ برنامج مخزن

اختبار ذاتي

ستجد في نهاية كل فصل من الفصول اختباراً ذاتياً قصيراً. الهدف من هذه الأسئلة الموضوعية (الأسئلة المتعدّدة الاختيارات) هو أن تتيح لك المجال للتأكد من مدى فهمك للمصطلحات الأساسية والمبادئ المقدّمة في كل فصل. أجب على الأسئلة وقارن أجوبتك بمفتاح الاجابات. إذا أخطأت في الاجابة على سؤال من الأسئلة فراجع المادة المتعلقة به. وإذا أخطأت في سؤالين أو ثلاثة أسئلة فيحسن قراءة الفصل ثانية.

1. الحقائق التي لا هيكل لها.

أ. معلومات
ب. معطيات
ج. خانات
د. عمليات معالجة

2. المعنى الذي تنسبه المخلوقات البشرية الى الحقائق:

أ. معلومات
ب. معطيات
ج. معرفة
د. هيكل

3. يعالج الكمبيوتر ____________ الى ____________.

أ. معلومات/معطيات
ب. معطيات/معلومات
ج. حقائق/معطيات
د. معلومات/حقائق.

4. تتدفّق المعطيات الى الكمبيوتر كـ ____________.

أ. خرج
ب. برنامج
ج. عملية
د. دخل

5. تتدفّق المعلومات من الكمبيوتر كـ ____________.

أ. خرج
ب. ذاكرة
ج. برنامج
د. دخل

6. تميّز ____________ الكمبيوتر عن الآلة الحاسبة.

أ. المعالج
ب. البرنامج المُخزّن
ج. الذاكرة
د. الخرج

7. تخزّن المعطيات وتعليمات البرنامج في ____________.

أ. المعالج
ب. جهاز خرج
ج. البرنامج المخزّن
د. الذاكرة

8. الجزء الذي يقوم فعلياً بمعالجة المعطيات في الكمبيوتر هو ____________.

أ. الذاكرة الرئيسية
ب. جهاز دخل
ج. المعالج
د. جهاز خرج

9. اللبنة الأساسية التي يُبنى عليها الكمبيوتر العصري هي ____________.

أ. الرقيقة
ب. الأنبوب
ج. الترانزستور
د. المفتاح

10. تُركّب الرقيقات عادة على ____________. تحتوي كل رقيقة نموذجية الالكترونيات اللازمة لمكّون من مكونات الكمبيوتر الرئيسية.

أ. الصفائح
ب. الألواح
ج. الاسطوانات
د. الحاملات

11. تُسمّى مكوّنات الكمبيوتر المادية مجتمعة بـ ____________.

أ. الأجهزة المحيطية
ب. الكيان المنطقي
ج. الألواح
د. الكيان المادي

12. تُسمَّى مجموعة البرامج مجتمعة بـ ____________.

أ. جهاز خزن مساعد
ب. الكيان المنطقي
ج. برامج مخزّنة
د. الكيان المادي

الاجابات

1.ب 2.أ 3.ب 4.د 5.أ 6.ب 7.د 8.ج 9.أ 10.ب 11.د 12.ب.

ربط المفاهيم

إن إدراك مفاهيم الكمبيوتر يقتضي من العلم أكثر من مجرد معرفة معاني بعض مصطلحات أساسية. لذلك، وبعد إجراء الاختبار الذاتي، ينتهي كل فصل بسلسلة أسئلة أكثر موضوعية ومنشّطة للذهن. وكثير من هذه الأسئلة سيطلب منك شرح مبدأ من المبادئ أو شرح الطريقة التي يعمل بها الكمبيوتر أو إيضاح مبدئين أو أكثر. وتتطلب أسئلة أخرى أن تقوم بربط المبادئ الواردة في فصلين أو أكثر من الفصول. وبعضها الآخر يتطلّب منك إجراء استنتاجات من المادّة المقدّمة في هذا الكتاب. لن تقترح أية أجوبة. إذا وجدت أي غموض في واحد من الأسئلة أو أكثر فاقرأ المادة ثانية. إذا لم تتمكّن من الاجابة عليها فاطلب المساعدة:

1. ما الفرق بين المعطيات والمعلومات.
2. اربط المصطلحين «معطيات» و«معلومات» مع المصطلحين دخل وخرج.
3. يميّز البرنامج المخزن الكمبيوتر عن الآلة الحاسبة. اشرح ذلك.
4. ما هو الكمبيوتر؟ لا تعطِ التعريف فقط بل اشرح معنى كل مصطلح فني يتضمّنه.
5. أوضح رسماً مبيناً فيه المكوّنات الأساسية لأنظمة الكمبيوتر النموذجية. اشرح بايجاز وظيفة كل مكون من المكونات.
6. اشرح الفرق بين الكيان المادي والكيان المنطقي.
7. اشرح بايجاز دورة الدخل/المعالجة/الخرج الأساسية الخاصة بالكمبيوتر.

2.

المعالج والذاكرة الرئيسية

مفاهيم أساسية

داخل الكمبيوتر

النظام العددي الثنائي

الذاكرة الرئيسية

- ☐ أجهزة الخزن المادية
- ☐ الخوينات والكلمات والعناوين
- ☐ ذاكرة العنونة
- ☐ ذاكرة الكتابة والقراءة

المعالج

- ☐ الدورات الآلية

داخل الكمبيوتر

درسنا في الفصل الأول دورة الدخل/المعالجة/الخرج، ووجدنا كيف يجري تخزين المعطيات والبرامج في الذاكرة الرئيسية وكيف يقوم المعالج، ضمن تحكم البرنامج المخزن، بتحويل المعطيات الخام الى معلومات. وكان الغرض من ذلك اعطاءك لمحة شاملة عن نظام الكمبيوتر.

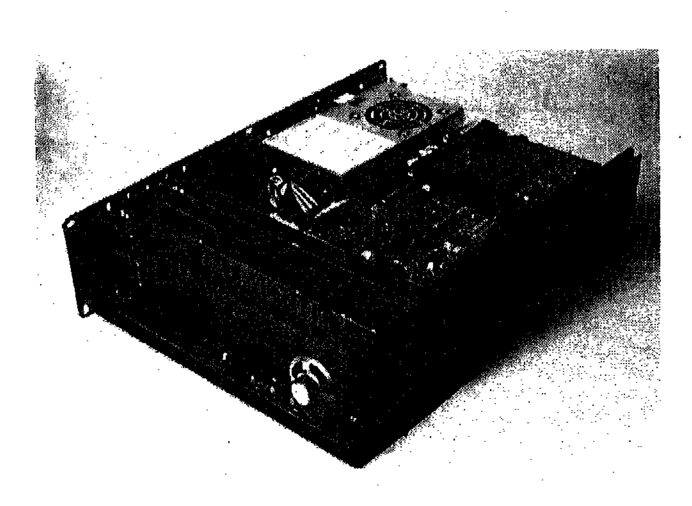
أمّا الخطوة التالية فهي تحليل كل من المكوّنات بشكل أكثر تفصيلاً. وسنبدأ بالكمبيوتر نفسه: اذا أزحنا غطاء كمبيوتر صغير وألقينا نظرة الى داخله (الشكل 2.1)، وجدنا بضعة ألواح دارات وبعض الكبلات والأسلاك. وتكمن فعالية الكمبيوتر الحقيقي في مجموعة الدارات الموجودة على الألواح. ولنلق نظرة أقرب. لكن قبل أن نفعل ذلك، علينا أن نأخذ بعين الاعتبار «لغة» الكمبيوتر: ثنائي.

النظام العددي الثنائي

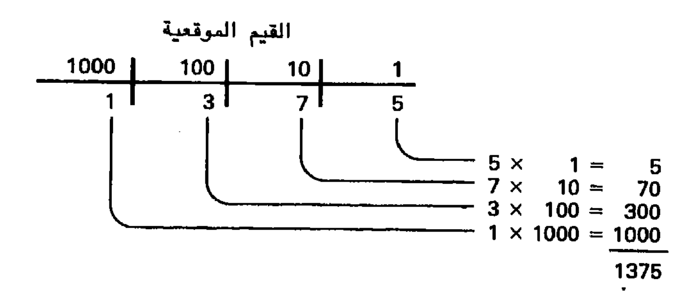
الكمبيوتر هو جهاز الكتروني يعمل بفعل نبضات تيار كهربائي موقتة توقيتاً دقيقاً. يتفاعل المعالج مع أنماط نابضة وغير نابضة. وتحتفظ الذاكرة بهذه الأنماط المتباينة (تشغيل/ايقاف). ويُعتبر النظام العددي **الثنائي** أفضل نظام لتمثيل هذه الأنماط لأنه يستخدم رمزين عدديين فقط: 0 و 1.

ما هو النظام العددي الثنائي بالضبط؟ إن النظام العددي بشكل عام هو رسم بياني لتمثيل القيم العددية. ففي النظام العشري، نستخدم مجموعات توافقية تتألف كل منها من عشرة رموز، 0 الى 9. إذ ان الأرقام وحدها ليست كافية غير ان مواقعها النسبية هامّة أيضاً. على سبيل المثال، من المعروف ان 03 هو ثلاث وان 30 هو ثلاثون بسبب موقع الرقم 3 النسبي. فلكل موقع قيمة، والقيم الموقعية (1 و10 و100 و1000) هي عوامل الرقم 10. تحدد قيمة عدد ما بضرب كل رقم بقيمته الموقعية ثم جمع تلك الحواصل (الشكل 2.2).

الشكل 2.1 يُبنَى الكمبيوتر من بضعة الواح دارات الكترونية.



الشكل 2.2 في النظام العددي العشري، القيم الموقعية هي عوامل الرقم عشرة. ولايجاد قيمة أي عدد، أضرب كل رقم بقيمته الموقعية واضف الحواصل.



في النظام العددي الثنائي تكون القيم الموقعية معامل مضاعف العدد اثنين 1 و2 و4 و8 و16...، (الشكل 2.3). وتتطلب عددين فقط وهما : 0 و1. ولا تزال قاعدة «الرقم - عدد المرات - الموقع - القيمة» قائمة، لكن كي تجد قيمة أي عدد من الأعداد، أضرب كل رقم بقيمته الموقعية واجمع الحواصل. لذلك، وكما تلاحظ في الشكل 2.3 ان العدد الثنائي 10 يساوي العدد العشري 2، بينما العدد 101 يساوي العدد العشري 5.

ونحن نستخدم الأعداد العشرية لأننا نجدها ملائمة (ربما لأن لدينا عشرة أصابع). ويستخدم الكمبيوتر الأعداد الثنائية لأنه يجدها ملائمة. أمّا الرقم الثنائي أو **الخوينة** (بت) هي وحدة الخزن الأساسية في الكمبيوتر. وتحتوي أجهزة الخزن المادية الخوينات (بتات) وتقوم أجهزة المعالجة بمعالجة هذه الخوينات، والكمبيوتر هو آلة قائمة على الكود الثنائي.

بالنسبة الى غالبية القراء إن تفهّم المبدأ الأساسي للنظام العددي الثنائي يُعتبر كافياً. أمّا القراء الآخرون، خصوصاً الطلبة الذين يرغبون في التخصّص في مجال الكمبيوتر فقد يحتاجون الى دراسة أعمق. هذا ويقدّم الملحق تفاصيل اضافية حول الأنظمة العددية وأشكال المعطيات والأنظمة الكودية.

الشكل 2.3 في النظام العددي الثنائي نجد ان القيم الموقعية هي عوامل العدد اثنين، وقاعدة «الرقم - عدد المرات - الموقع - القيمة» لا تزال قائمة.

القيم الموقعية
32 16 8 4 2 1
1 0
0 × 1 = 0
1 × 2 = 2
2
1 0 1
1 × 1 = 1
0 × 2 = 0
1 × 4 = 4
5

الذاكرة الرئيسية

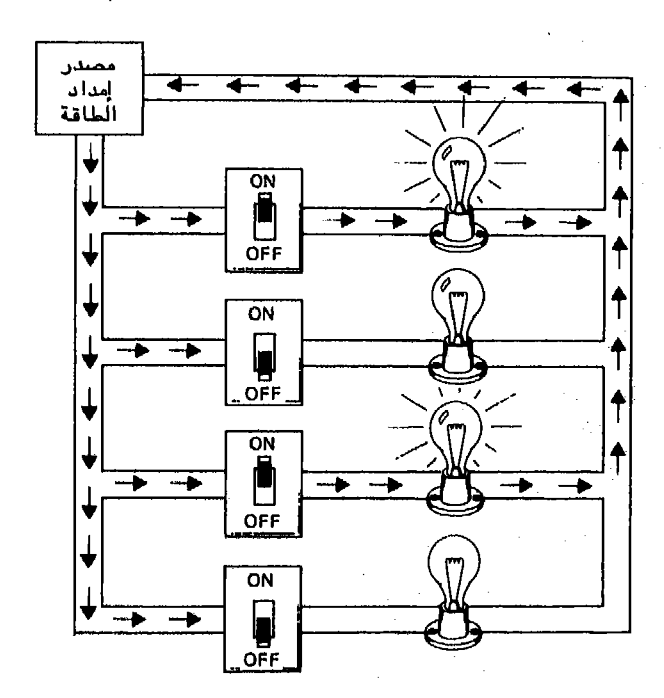
أجهزة الخزن المادية

لنتأمّل مجمع المفاتيح والأضواء المصوّرة في الشكل 2.4، فنجد ان كل مفتاح منها هو بمثابة جهاز ميكانيكي بسيط يتولى إحداث إحدى الحالتين «اشعال النور وإطفاوه Off/on». فلنضبط المفاتيح الآن ونوصل التيار الكهربائي. أي المصابيح تضيء؟ ومن الواضح ان المصابيح التي ستضاء هي تلك التي ضبطناها. وإذا أعدنا إجراء التجربة، فإن المصابيح ذاتها ستضاء. واذا غيّرنا مواضع المفاتيح فإن مصابيح أخرى ستضاء. والمفتاح هو جهاز خزن أولي أو ذاكرة يكون فيه نمط الاشعال On ممثلاً بخوينة 1 ونمط الاطفاء Off ممثلاً بخوينة 0.

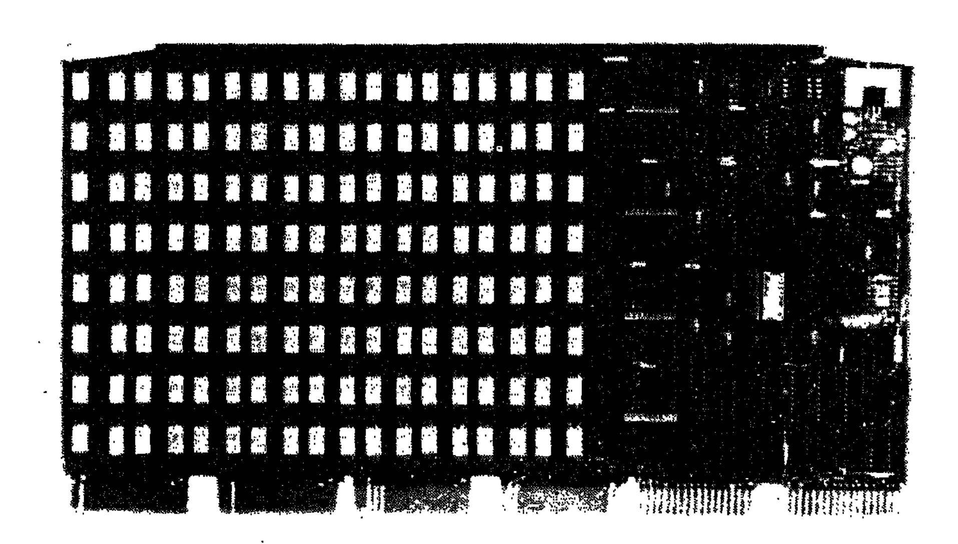
يجب ان نميّز بكل عناية الفرق بين الجهاز المادي (المفتاح) وقيمته (0 أو 1). فالمفتاح جهاز مادي، وتهيئته تُعتبر الكيان المنطقي أو المعطيات. يمكن تغيير القيمة بسهولة بمجرد نقرة صغيرة على المفتاح. أمّا القيمة، الكيان المنطقي أو المعطيات، فليست مستمرة بينما الكيان المادي (المفتاح) فهو دائم.

إن أي جهاز يمكنه أن يتولّى إحدى الحالتين ويحتفظ بها يكون جهاز خزن كمبيوتري محتمل. تستخدم معظم أجهزة الكمبيوتر الحديثة **ذاكرة الدارات المتكاملة** (الشكل 2.5). وقد يحتوي كمبيوتر صغير على ذاكرة ذات سعة كافية لخزن آلاف الخوينات، وقد تستوعب آلة كبيرة ملايين الخوينات. الفرق اذاً هو في قدرة السعة لا في الوظيفة.

الشكل 2.4 مفتاح المصباح الكهربائي هو مثل بسيط عن أجهزة الخزن او الذاكرة.



الشكل 2.5 لوح ذاكرة بدارات متكاملة.



الخانات والكلمات

بإمكان خوينة واحدة أن تحتفظ بـ 0 أو 1. لكن عموماً، يمكن تخيّل محتويات الذاكرة كمجموعات خوينات وليس كخوينات منفردة (للأسباب نفسها التي تجعلنا نركّز على الكلمات والجمل وليس على الحروف المنفردة عندما نقرأ رواية من الروايات). ومع ان الخوينة ما زالت وحدة الخزن الأساسية، فإن أجهزة الكمبيوتر تعمل على معالجة الخانات أو الكلمات.

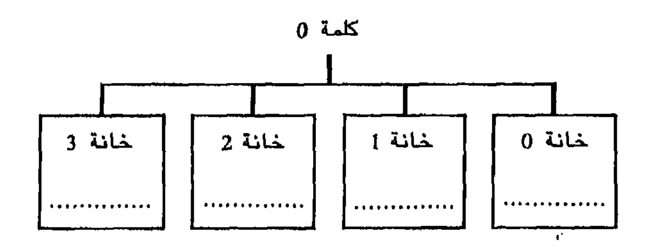
وتحتوي **الخانة** على بضع خوينات (عادة ثمان) لتمثيل رمز واحد. وهناك نظامان للأكواد الثنائية شائعا الاستخدام لتمثيل الرموز في الكمبيوتر: كود «الآسكي» (نظام الشيفرة الأمريكي القياسي لتبادل المعلومات) وكود الأبسيديك (الكود العشري الثنائي التكويد الموسع لتبادل المعطيات) (الشكل 2.6). مثلاً، إن الحرف اللاتيني الكبير A في نظام الآسكي بالكود الثنائي يمثل 01000001. وبإمكان خانة واحدة (مؤلفة من ثماني خوينات) استيعاب ذلك النمط من الخوينات داخل الذاكرة الرئيسية. وبوجه عام، تتسع كل خانة لرمز مكود واحد (ان كان ذلك حرفاً أو رقماً أو علامة ترقيم).

إن الخانات جيدة لخزن الرموز لكنها أصغر من أن تتسع لخزن عدد ذي معنى. إن معظم أجهزة الكمبيوتر تستطيع أن تعالج مجموعة من الخانات تسمّى **كلمة**. كما أن بعض أجهزة الكمبيوتر الصغيرة تتضمّن كلمات ذات 8 خوينات. وتعمل بعض أجهزة الكمبيوتر الأقوى الأخرى باستخدام كلمات ذات 16 خوينة (خانتان) و32 خوينة (أربع خانات) وحتى 64 خوينة (ثماني خانات). لذلك ينتج لدينا تسلسلاً هرمياً في الذاكرة (الشكل 2.7). ووحدة الخزن الأساسية في الكمبيوتر هي الخوينة. وتقع الخوينات في مجموعات لتشكّل الخانات وهي بدورها تجمع لتشكيل كلمات. وفي إحدى الاستخدامات التطبيقية قد تستخدم كلمة ما للاحتفاظ بعدد ثنائي، وفي استخدام آخر، قد تحتوي خانات الكلمة على رموز منفردة أو تعليمة برنامج.

الشكل 2.6 غالباً ما يستخدم النظامان الكوديان آسكي ASCII وابسيديك EBCDIC لتمثيل الرموز في الكمبيوتر. أمّا الأنماط الخوينية الأخرى غير المبيّنة فتُستخدم في تمثيل علامات الترقيم والرموز الخاصة الأخرى.

ASCII-8	EBCDIC	سمة
0100 0001	1100 0001	A
0100 0010	1100 0010	B
0100 0011	1100 0011	C
0100 0100	1100 0100	D
0100 0101	1100 0101	E
0100 0110	1100 0110	F
0100 0111	1100 0111	G
0100 1000	1100 1000	H
0100 1001	1100 1001	I
0100 1010	1101 0001	J
0100 1011	1101 0010	K
0100 1100	1101 0011	L
0100 1101	1101 0100	M
0100 1110	1101 0101	N
0100 1111	1101 0110	O
0101 0000	1101 0111	P
0101 0001	1101 1000	Q
0101 0010	1101 1001	R
0101 0011	1110 0010	S
0101 0100	1110 0011	T
0101 0101	1110 0100	U
0101 0110	1110 0101	V
0101 0111	1110 0110	W
0101 1000	1110 0111	X
0101 1001	1110 1000	Y
0101 1010	1110 1001	Z
0011 0000	1111 0000	0
0011 0001	1111 0001	1
0011 0010	1111 0010	2
0011 0011	1111 0011	3
0011 0100	1111 0100	4
0011 0101	1111 0101	5
0011 0110	1111 0110	6
0011 0111	1111 0111	7
0011 1000	1111 1000	8
0011 1001	1111 1001	9

الشكل 2.7 في ذاكرة الكمبيوتر الرئيسية، تُجمع الخوينات لتشكيل الخانات، وتُجمع الخانات بدورها لتشكيل الكلمات. يوضح الرسم التالي كلمة مؤلفة من 32 خوينة أو 4 خانات.



من المهمّ التنبّه الى ان معطيات الرموز والمعطيات العددية يختلف بعضها عن البعض الآخر. وتمثّل الرموز بكود، وكل كود مستقل بذاته. إن مفهوم القيمة الموقعية لا علاقة له، فيضرب كل رقم من الارقام في قيمته الموقعية ثم جمع الحواصل يؤدّي الى نتيجة لا معنى لها. وبالمقابل، فان الخوينات التي تشكل عدداً ما تكوّد في مواقع نسبية معينة، كل منها له قيمة موقعية واحدة ـ 8، 4، 2، 1 وهكذا؛ فان قانون «الرقم ـ عدد المرات ـ الموقع ـ القيمة» يصلح مع الأعداد ولا يُستخدم مع الرموز. (أنظر الملحق لمزيد من الشرح).

ذاكرة العنونة

يحتوي الميكروكمبيوتر النموذجي على 128000 خانة (128 كيلوبايت☆) أو أكثر من الخانات أو الكلمات، بينما يمكن أن يحتوي كمبيوتر رئيسي على ملايين من الخانات. وقد يخزّن عنصر معطيات معين في أي منها. اذا احتاج المعالج عنصر معطيات معيناً فكيف يقوم بإيجاد الخانة أو الكلمة التي تحتويه؟

يحدّد لكل وحدة خزن ماديّة **عنوان** وحيد. أمّا بالنسبة لمعظم أجهزة الكمبيوتر فترقم الخانات أو الكلمات بشكل متتابع ـ بدءاً بـ2، 1، 0 وهلمّ جرا. ويقوم المعالج بنيل موقع ذاكرة معين بالاشارة الى عنوانه. على سبيل المثال، اذا احتاج المعالج الى خزن المعطيات في خانة 1048، فيسأل الذاكرة أن تزوّده بمحتويات الخانة 1048. وبما ان هناك خانة 1048 واحدة، فإن المعالج يحصل على المعطيات الصحيحة. ويُحسب على الكمبيوتر، فإن الخانات والكلمات هي وحدات الذاكرة الأساسية التي يمكن نيلها. وتنتقل المعطيات ما بين المعالج والذاكرة الرئيسية كخانة أو ككلمة في كل مرة.

ذاكرة الكتابة والقراءة

يتمّ نيل موقع ما في **الذاكرة الرئيسية** عن طريق عنوانه. وغالباً تقرأ محتويات الذاكرة فقط. وعندما تقرأ الذاكرة، لا تتغير محتوياتها. مثلاً، تصوّر أن الخانة رقم 42 تحتوي على الحرف A، وإذا قرأ المعالج الخانة 42 سيبقى الحرف A فيها. غير ان عملية الكتابة في الذاكرة الرئيسية تؤدّي الى إتلاف المحتويات السابقة. وإذا كتب المعالج الحرف X في الخانة 42 تستبدل القيمة القديمة بالقيمة الجديدة وتفقد A الى الأبد.

تتألف الذاكرة الرئيسية في معظم أجهزة الكمبيوتر من **ذاكرة نيل عشوائي أو RAM**. وفي وسع المبرمج (طبعاً باستخدام برنامج ما) القراءة أو الكتابة في ذاكرة النيل العشوائي. ويمكن خزن معطيات الدخل في ذاكرة RAM متلفة محتويات خانات أو كلمات مختارة سابقة. وحال إدخال المعطيات، يمكن قراءتها ومعالجتها بواسطة المعالج، وتكتب النتائج في مواقع أخرى في الذاكرة. وفي النهاية، يمكن إرسال المحتويات الى جهاز خرج. وعند انتهاء البرنامج يمكن نسخ برنامج جديد في ذاكرة النيل العشوائي ويمحى البرنامج القديم في هذه العملية. ومن السهل تغيير محتويات ذاكرة النيل العشوائي.

وعادة نجد ان مرونة ذاكرة النيل العشوائي هي مزية يتمتع بها، ولكن أحياناً قد تسبب بعض المشاكل. خذ على سبيل المثال، طرفيات الصرف المباشر المستخدمة في

☆ إن اللاحقة «ك» K والتي تمثّل كيلو تعني 1024. وهكذا نجد أنه اذا كانت سعة ذاكرة الكمبيوتر 128ك خانة، فإنه يحتوي على 131,072 موقع ذاكرة فعلي.

كثير من البنوك، والتي يتمّ التحكّم بها عن طريق أجهزة كمبيوتر صغيرة والتي بدورها تتحكّم بها برامج معينة: فقد ينجح مبرمج ذكي في تغيير برنامج من هذه البرامج لاتاحة الوصول دون عائق الى حسابات معينة. ولا ضرورة لذكر ان البنوك لا تسمح بإحداث أي تغيير في هذه الحسابات. إن البنوك بحاجة الى برنامج يمكن أن يقرأه المعالج ولا يمكن إجراء تغيير عليه. لذلك تُخزّن مثل هذه البرامج في **ذاكرات القراءة فقط ROM.** هناك مثال جيد آخر عن البرامج المرتكزة الى ذاكرات القراءة فقط هو مترجم لغة البيسيك الموجود في العديد من أجهزة الميكروكمبيوتر. وسنبحث المترجمات في الفصل 7. والمصطلح ROM على ما يدلّ عليه اسمه هو ذاكرة «مستمرّة» يمكن قراءتها ولكن لا يمكن كتابتها.

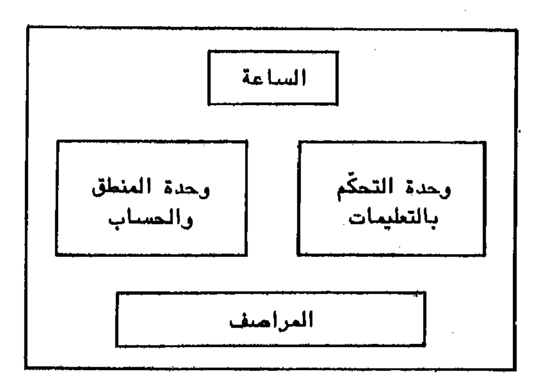
المعالج

المعالج الذي غالباً ما يُعرف **بوحدة المعالجة المركزية CPU** أو **بالمعالج الرئيسي،** هو المكوّن الذي يعالج المعطيات ويتحكّم بها. لا يستطيع المعالج إنجاز اية مهمة بدون برنامج يوفّر له التحكّم بالمهام الموكولة اليه. إن مصدر الذكاء في أي كمبيوتر هو الكيان المنطقي لا الكيان المادي. كما يعمل المعالج على معالجة المعطيات المخزّنة في الذاكرة الرئيسية تحت رقابة البرنامج المُخزّن فيها.

الشكل 2.8 تتكوّن التعليمة من كود عملية ومعامل واحد أو أكثر. يوجّه كود العملية الكمبيوتر للقيام بمهام معيّنة. إن المعامل أو المعاملات تتعرّف على عناوين عناصر المعطيات المراد معالجتها.

كود العملية	المعامل
ADD	1000,1002

الشكل 2.9 يشتمل المعالج على أربعة مكوّنات رئيسية.



ان البرنامج هو الذي يتحكّم في المعالج. ويتألف البرنامج من سلسلة **تعليمات**. وتتكوّن كل تعليمة من مجموعة خوينات تطلب من الكمبيوتر إنجاز واحدة من وظائفه الأساسية: الجمع أو الطرح أو الضرب أو القسمة أو المقارنة أو النسخ أو بدء الدخل أو الخرج..وتتألف كل تعليمة من جزئين؛ **كود العملية ومعامل** واحد أو أكثر (الشكل 2.8). ويبلغ كود العملية الكمبيوتر بنوع الوظيفة التي يقوم بأدائها (مثل اجمع، اطرح، قارن)، وتحدّد المعاملات مواقع الذاكرة التي ستسهم في أداء العملية. مثلاً إن التعليمة الموضحة في الشكل 2.8 تطلب من الكمبيوتر أن يجمع محتويات مواقع الذاكرة 1000 و1002.

يحتوي المعالج على أربعة مكوّنات رئيسية (الشكل 2.9): ساعة ووحدة تحكّم بالتعليمات ووحدة المنطق والحساب ومجموعة من المراصف. وتولد **الساعة** نبضات تيّار موقتة توقيتاً دقيقاً لتزامن مكوّنات المعالج الأخرى. وتحدد **وحدة التحكّم بالتعليمات** مواقع التعليمات اللاحقة المُراد تنفيذها، وتقوم **باستحضارها** من الذاكرة الرئيسية. وتُنفّذ **وحدة المنطق والحساب** تلك التعليمة. وهي تتألف من الدارات التي تقوم بعمليات الجمع أو الطرح أو الضرب أو القسمة أو المقارنة أو النسخ أو إحداث الدخل أو الخرج ـ ويطلق على ذلك **مجموعة التعليمات** الخاصة بالكمبيوتر.. أمّا المراصف فهي أجهزة (أماكن) خزن موقتة تحتفظ بمعلومات تحكم ومعطيات رئيسية ونتائج وسيطة.

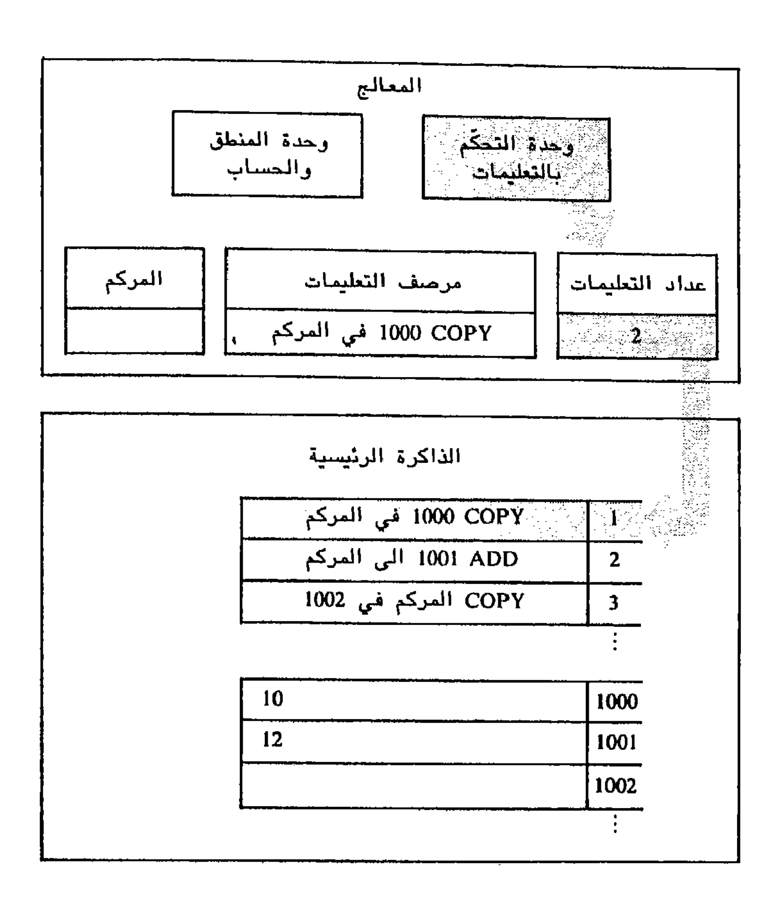
الدورات الآلية

كيف، بالضبط، تعمل مكوّنات الكمبيوتر الداخلية معاً لتنفيذ تعليمة ما؟ تعالوا نستخدم نموذجاً مبسّطاً لنظام كمبيوتر (الشكل 2.10أ) لتوضيح بضع **دورات آلية**. ابدأ بالمعالج. بالاضافة الى الساعة (ليست موجودة في هذا الشكل)، فإنه يتضمّن وحدة تحكّم بالتعليمات، ووحدة منطق وحساب، وعدة مراصف بما في ذلك عداد للتعليمات ومرصف تعليمات ومرصف عمل يُسمّى بالمركم. إن الذاكرة الرئيسية، المكّون الآخر من مكوّنات الكمبيوتر الرئيسية، تحتفظ بتعليمات البرنامج وقيم المعطيات، والأعداد التي تتقدّم التعليمات وقيم المعطيات فهي عناوين الذاكرة الرئيسية.

تبدأ العملية عندما تولد الساعة نبضة تيار تنشط وحدة التحكّم بالتعليمات والتي وظيفتها تقرير ما ستقوم به الآلة فيما بعد. يضبط الكمبيوتر بواسطة تعليمات البرنامج وعليك أن تتذكر أن التعليمات مخزونة في الذاكرة الرئيسية. كما يوجد عنوان التعليمة التالية المُراد تنفيذها في عداد التعليمة (الشكل أ 2.10). وتبحث وحدة التحكّم بالتعليمات في عداد التعليمات وتجد العنوان وتستحضر التعليمة التالية وتضعها في مرصف التعليمات (الشكل ب 2.10). قد تستهلك عملية استحضار تعليمة من الذاكرة فترة من الزمن مما يتيح الفرصة لوحدة التحكم بالتعليمات لدفع عداد التعليمات ليشير الى التعليمة التالية (الشكل ب 2.10).

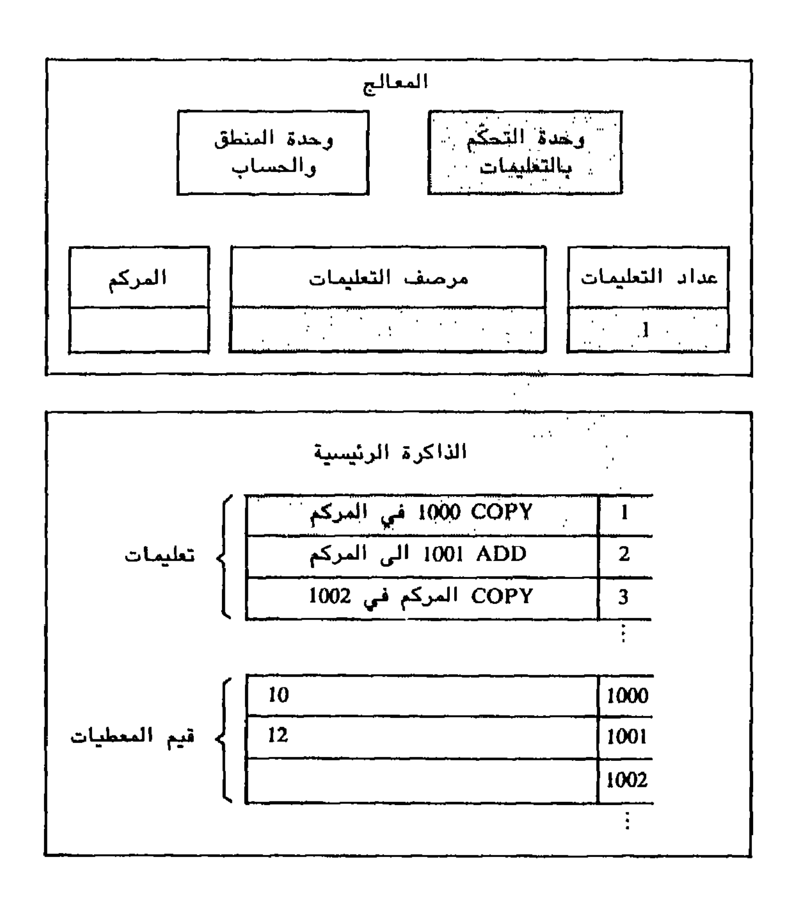
تولد الساعة على فترات زمنية دقيقة نبضات تيار كهربائي أخرى. فتنشط هذه النبضات وحدة المنطق والحساب التي تقوم، بدورها، بتنفيذ التعليمة المخزونة في مرصف التعليمات (الشكل 2.10ج). لاحظ انه بعد الانتهاء من تنفيذ تعليمة ما تنسخ قيمة المعطيات من الذاكرة الرئيسية الى المرصف.

الشكل 2.10 ينفّذ الكمبيوتر تعليمة واحدة في أثناء كل دورة آلية واحدة.



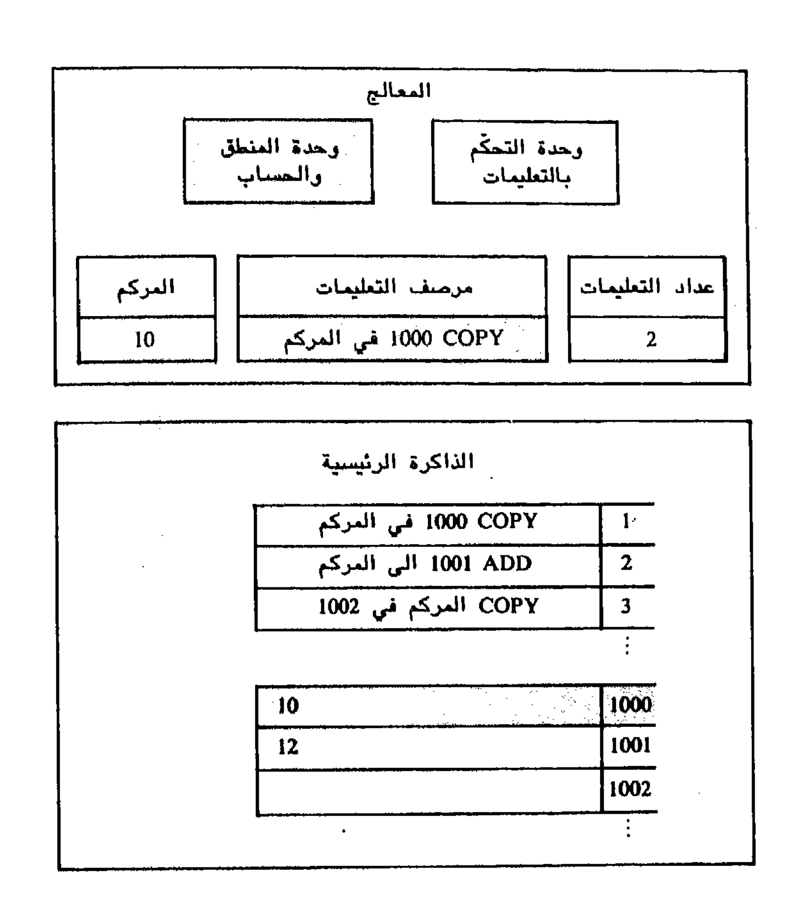
. يبدأ مثلنا بالذاكرة الرئيسية التي تحتفظ بكلّي تعليمات البرنامج والمعطيات. ويشير مرصف التعليمات نحو التعليمة الأولى المُراد تنفيذها.

الشكل 2.10



ب. استجابة لأمر استحضار تولده وحدة التحكّم بالتعليمات، تنسخ اول تعليمة من الذاكرة وتُخزّن في مرصف التعليمات. لاحظ أن عدّاد التعليمات يشير الى التعليمة *اللاحقة*.

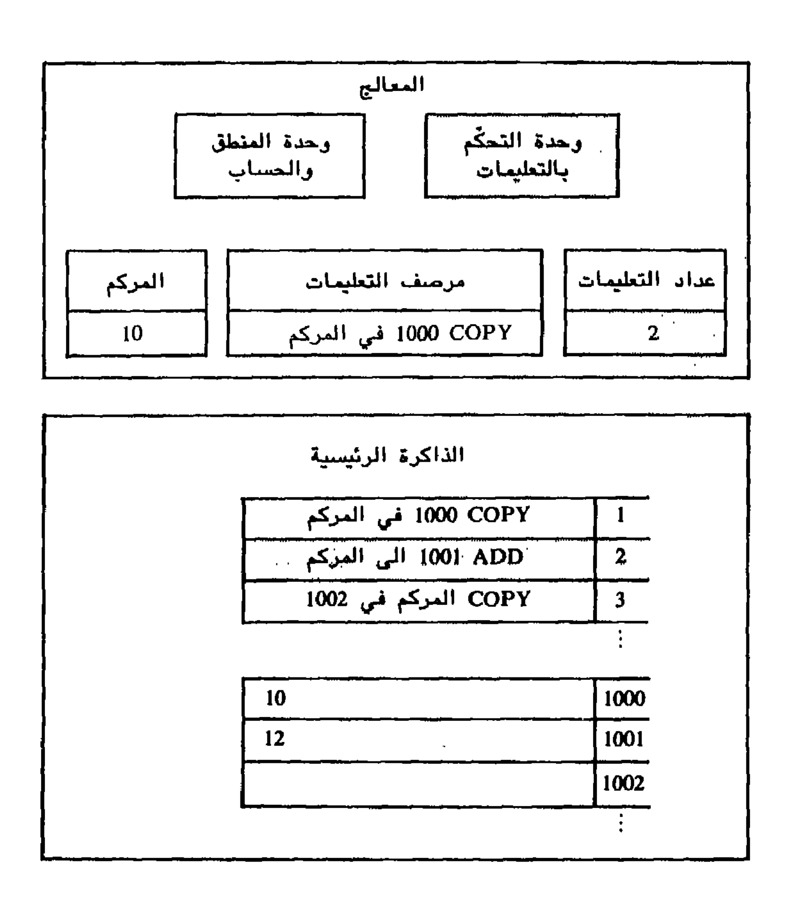
الشكل 2.10



ج. في الخطوة التالية، تتولّى وحدة المنطق والحساب زمام الأمور وتنفّذ الأمر في مرصف التعليمات. وبذلك تنسخ قيمة المعطيات من الذاكرة الرئيسية وتحمل في المركم.

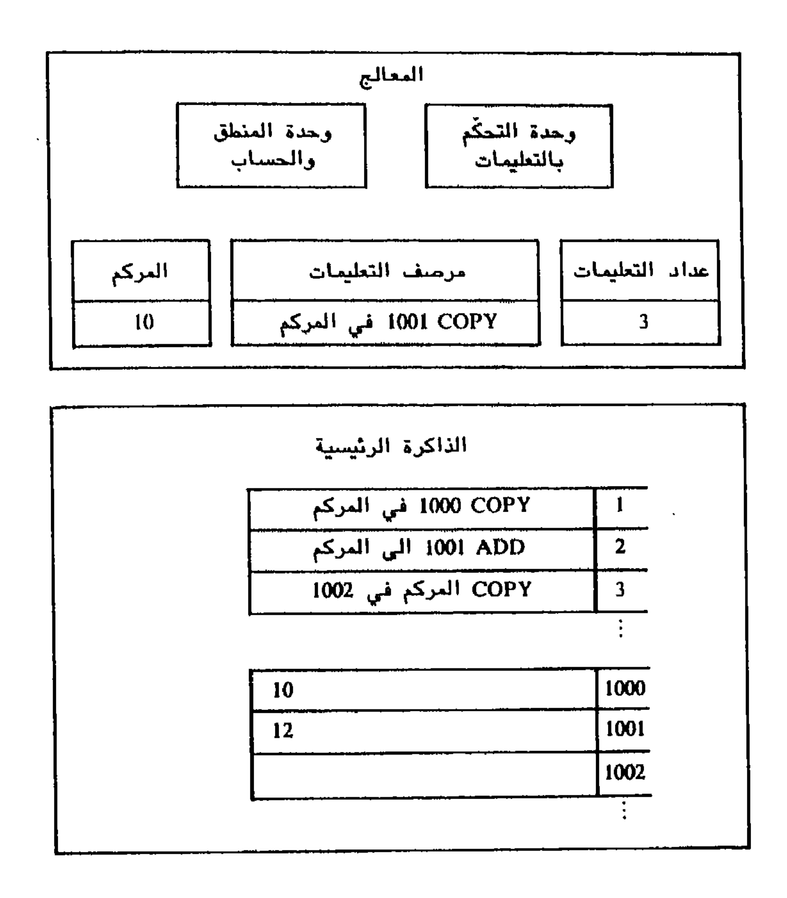
مرة ثانية، تتكّ الساعة وهكذا تعود الي وحدة التحكّم بالتعليمات حيث تبدأ الدورة الآلية التالية (الشكل د 2.10) مشيراً الى عداد التعليمات، تقوم وحدة التحكم بالتعليمات باستحضار التعليمة اللاحقة وتنسخها في مرصف التعليمات (الشكل هـ-2.10). كما يجب أن تلاحظ الآن ان مرصف التعليمات يشير الى التعليمة اللاحقة.

الشكل 2.10



د. عندما تبدأ الدورة التالية، تستعيد وحدة التحكّم بالتعليمات قدرة التحكّم ثانية وتتطلّع الى مرصف التعليمة ليحصل على عنوان التعليمة اللاحقة.

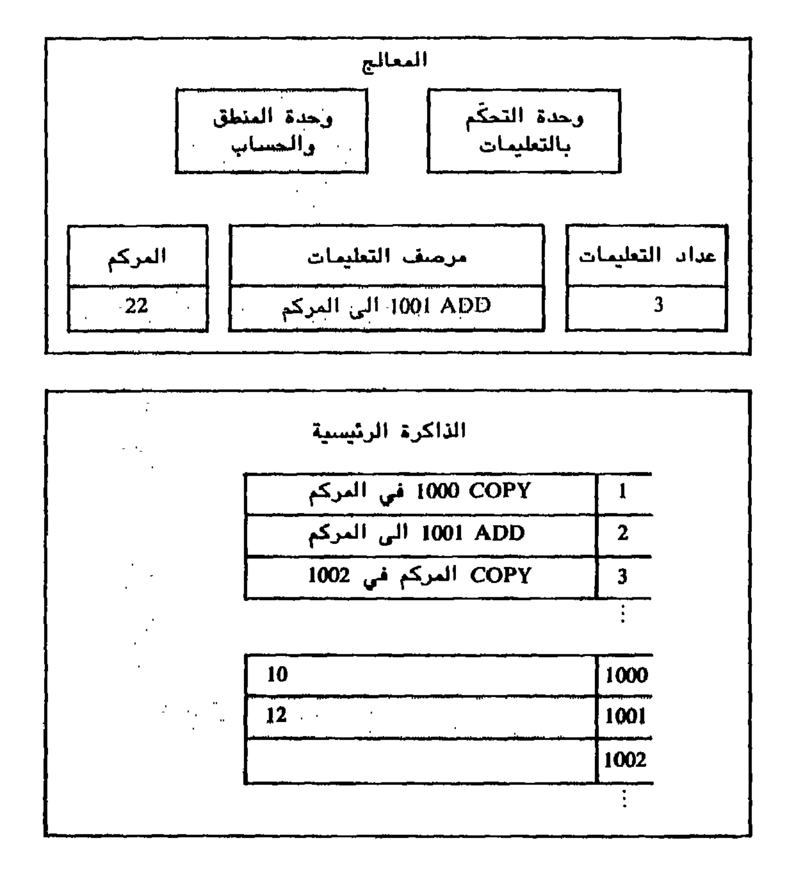
الشكل 2.10



هـ. تستحضر التعليمة التالية وتنسخ في مرصف التعليمة. لاحظ أن عدّاد التعليمة يشير الى التعليمة اللاحقة.

وتتكّ الساعة، وتحصل وحدة المنطق والحساب على قدرة التحكّم وتنفّذ التعليمة في مرصف التعليمة (الشكل و2.10)، وبذلك تُضاف قيمة معطيات من الذاكرة الى المركم.

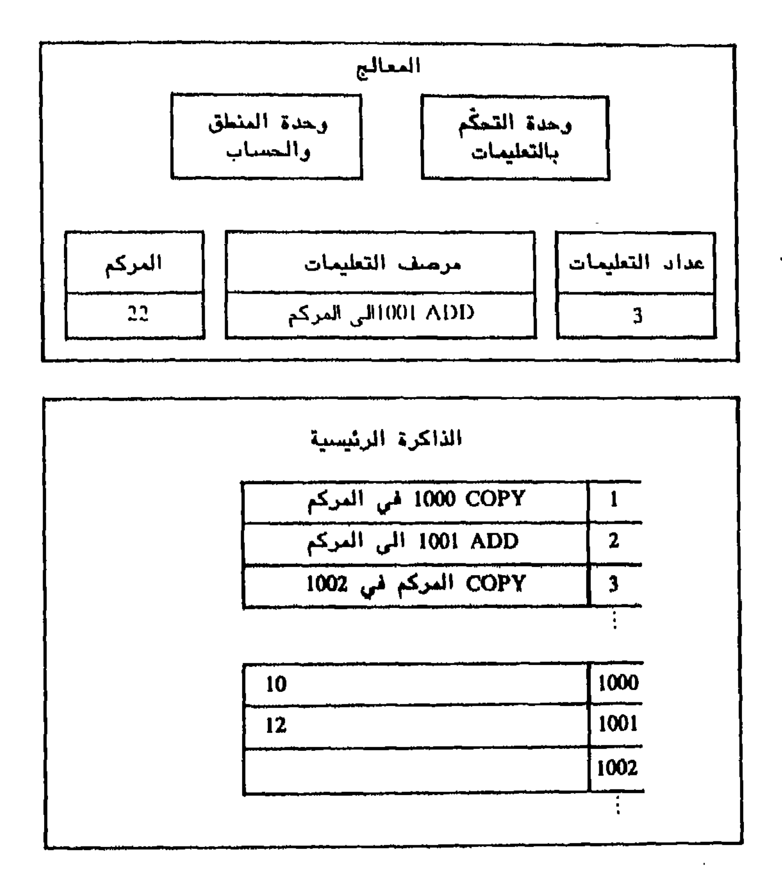
الشكل 2.10



و. تنفّذ وحدة المنطق والحساب التعليمة في المرصف. وتتمّ إضافة قيمتين ويوضَع المجموع في المركم.

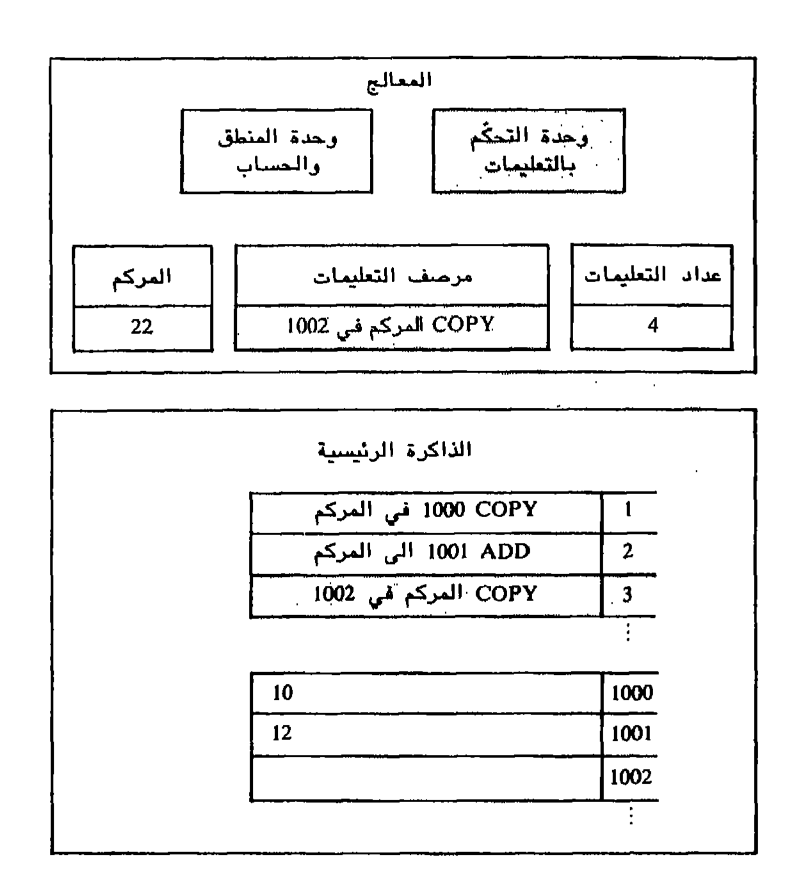
تعود بنا نبضة الساعة التالية الى وحدة التحكّم بالتعليمات. وكما جرى سابقاً، يشير عدّاد التعليمة الى التعليمة التالية (ز2.10) ويجري استحضار ونسخ هذه التعليمة في مرصف التعليمات (الشكل ح2.10). حسبما جرى سابقاً، تمنح نبضة الساعة التالية قدرة التحكّم لوحدة المنطق والحساب التي تقوم بتنفيذ التعليمة (الشكل ط2.10) وتخزّن المجموع في الذاكرة الرئيسية.

الشكل 2.10



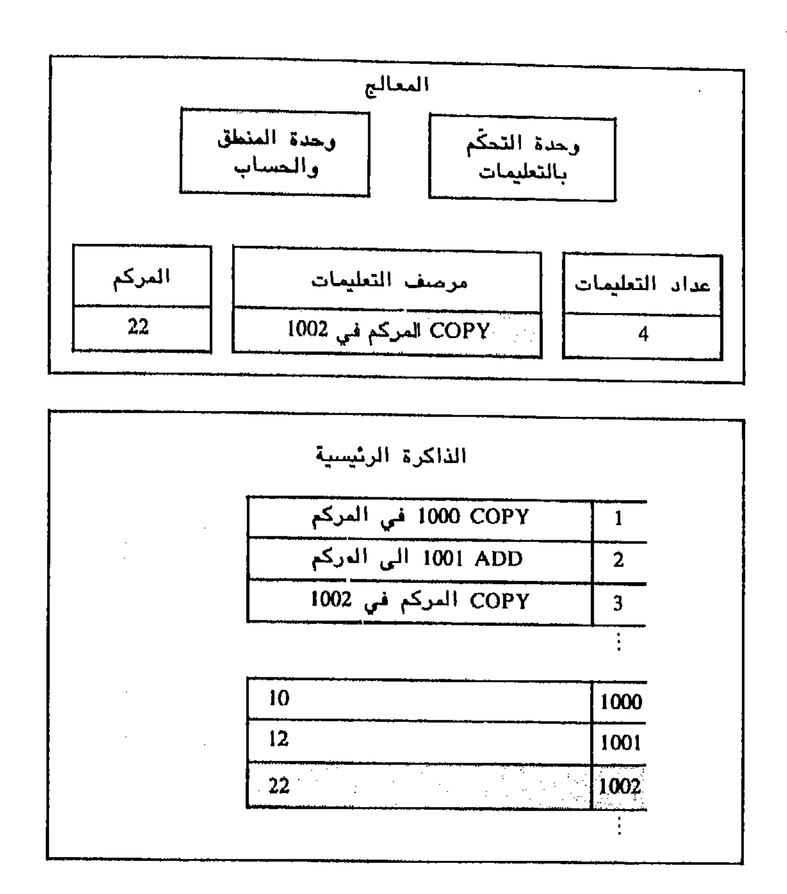
ز. عندما تبدأ الدورة التالية، تتوقّع وحدة التحكّم بالتعليمات ان تحصل من عداد التعليمات على عنوان التعليمة التالية.

الشكل 2.10



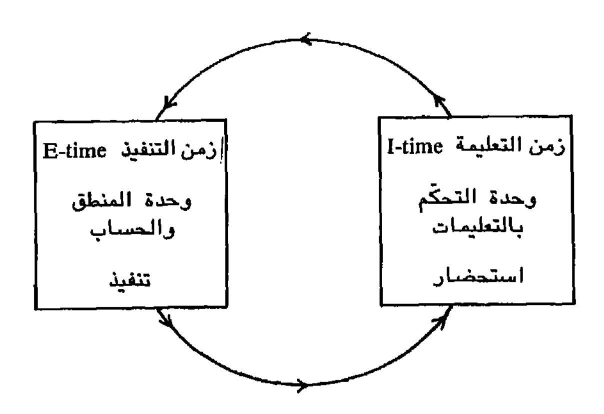
ح. يتمّ استحضار ونسخ التعليمة التالية في مرصف التعليمات. لاحظ بأن عداد التعليمة يشير الى التعليمة التالية.

الشكل 2.10



ط. تقوم وحدة المنطق والحساب بتنفيذ التعليمة في مرصف التعليمات. ثم يتمّ تخزين النتيجة في الذاكرة الرئيسية ثانية.

الشكل 2.11 تتكرّر الدورة الآلية الأساسية مرات ومرات الى أن تكون قد نفّذت جميع التعليمات في البرنامج.



يتمّ استحضار التعليمة عن طريق وحدة التحكّم بالتعليمات في أثناء «**زمن التعليمة**» **I-time**، وتقوم وحدة المنطق والحساب بتنفيذها في أثناء **زمن التنفيذ E-time** (الشكل 2.11). ويجري تكرار هذه العملية مرات عديدة الى أن ينتهي البرنامج. وتدفع الساعة بهذه العملية الى الأمام مولّدة نبضات تيار كهربائي على فترات زمنيّة منتظمة. إن المعدل الذي تولد فيه نبضات الساعة هو الذي يحدّد سرعة تشغيل الكمبيوتر. ويُقاس زمن الساعة في كثير من أجهزة الكمبيوتر الحديثة بالنانوثواني (واحد على بليون من الثانية)، وبذلك تتمكّن الآلة من تنفيذ ملايين التعليمات في الثانية الواحدة.

الخلاصة

حتى هذه اللحظة، عرفنا ان الكمبيوتر هو آلة تعالج المعطيات (المخزّنة في الذاكرة الرئيسية) في صيغة معلومات تحت اشراف برنامج مخزون. كما اننا عرفنا ان الكمبيوتر (داخلياً) هو آلة ثنائية، وهكذا يجب خزن المعطيات وتعليمات البرنامج بشكل عددي ثنائي. ويتم تمثيل الرموز بالكود الثنائي، وتخزن الأعداد في شكل أعداد ثنائية، تكون كل قيمة موقعية خاصة بخوينة من الخوينات ذات أهمية. وتنقسم الذاكرة الرئيسية في كمبيوتر ما الى خانات أو كلمات أو كليهما معاً (وذلك يتوقف على نوع النظام)، ويخصّص لكل وحدة من وحدات الخزن هذه عنوان. وباستعمال هذا العنوان، يمكن للمعالج قراءة أو كتابة الخانات أو الكلمات المختارة.

ويشتمل المعالج على ساعة ووحدة تحكم بالتعليمات، ووحدة المنطق والحساب والمراصف. وحالما يتمّ خزن برنامج ما في الذاكرة الرئيسية يمكن للمعالج أن يبدأ بتنفيذه. وفي أثناء «زمن التعليمة» تقوم وحدة التحكم بالتعليمات باستحضار التعليمة من الذاكرة الرئيسية. وفي أثناء «زمن التنفيذ» تقوم وحدة المنطق والحساب بتنفيذها. وتقوم النبضات الالكترونية الموقتة توقيتاً دقيقاً التي تولدها الساعة بدفع دورة الآلة الأساسية هذه.

ما هو مصدر المعطيات؟ وكيف يدخل برنامج ما في الذاكرة الرئيسية؟ ستتمّ الاجابة على هذه الأسئلة في الفصلين اللاحقين.

مصطلحات أساسية

- ☐ عنوان
- ☐ وحدة المنطق والحساب
- ☐ ثنائي
- ☐ خوينة
- ☐ خانة
- ☐ وحدة المعالجة المركزية
- ☐ ساعة
- ☐ زمن التنفيذ
- ☐ استحضار
- ☐ تعليمة
- ☐ وحدة التحكم بالتعليمات
- ☐ مجموعة تعليمات
- ☐ زمن التعليمة
- ☐ ذاكرة دارة متكاملة
- ☐ الدورة الآلية
- ☐ الذاكرة الرئيسية
- ☐ المعالج الرئيسي
- ☐ معامل
- ☐ كود عملية
- ☐ معالج
- ☐ ذاكرة نيل عشوائي (RAM)
- ☐ مرصف
- ☐ ذاكرة قراءة فقط ROM
- ☐ كلمة

اختبار ذاتي

1. القيم الموقعية في النظام العددي الثنائي هي عوامل ________

أ. عشرة ج. ثمانية
ب. اثنان د. ستة عشر

2. المفتاح المادي هو ________ وضبطه يُعتبر ________

أ. كيان منطقي/كيان مادي ج. كيان مادي/معطيات
ب. كيان مادي/كيان منطقي د. معطيات/معلومات

تحتوي ________ عدداً كافياً من الخوينات لخزن رمز واحد.

أ. الكلمة ج. الخوينة
ب. الخانة د. الرقيقة

4. ________ هو/هي مجموعة من ________.

أ. خوينة/خانات ج. كلمة/خانات
ب. خانة/كلمات د. رمز/خانات

5. تمثّل الرموز داخل الكمبيوتر بـ ________

أ. كود ثنائي ج. مراصف
ب. كلمات د. قيم موقعية

6. يعمل قانون «الرقم ـ عدد المرات ـ الموقع ـ القيمة» وفقاً لـ________ وليس لـ________.

أ. أعداد/رموز
ب. خوينات/خانات
ج. خانات/كلمات
د. رموز/أعداد

7. يتمّ نيل موقع من مواقع الذاكرة عن طريق ________

أ. العنوان
ب. القيمة
ج. المحتوى
د. المرصف

8. عندما ________ الذاكرة، لا يتغيّر محتواها.

أ. تكتب
ب. يتم نيل
ج. تقرأ
د. تتغير دائماً

9. عندما ________ الذاكرة، يتغيّر محتواها.

أ. تكتب
ب. يتم نيل
ج. تقرأ
د. تتغير دائماً

10. يمكن للمبرمج قراءة وكتابة ________.

أ. ذاكرة القراءة فقط
ب. ذاكرة النيل العشوائي
ج. الكلمات
د. الخوينات

11. ما هو نوع الذاكرة التي يمكن قراءتها فقط

أ. مراصف
ب. ذاكرة القراءة فقط
ج. ذاكرة النيل العشوائي
د. الكلمات

12. يقوم المعالج باستحضار وتنفيذ ________.

أ. نبضات الساعة
ب. المعطيات
ج. التعليمات
د. المعلومات

13. ذلك الجزء من التعليمة الذي يوعز للمعالج بما سيؤدّيه من مهام هو ________

أ. المعامل
ب. النبضة
ج. العبارة
د. كود العملية

14. أي من أجزاء التعليمة يتعرّف على مواقع الذاكرة التي ستشترك في عملية ما؟

أ. كود العملية
ب. النبضة
ج. العبارة
د. المعامل (المعاملات)

15. إن مكوّنات المعالج تزامن بواسطة ________

أ. التعليمات
ب. المعطيات
ج. المعلومات
د. نبضات الساعة

16. أي من مكوّنات المعالج يقوم باستحضار التعليمة التالية من الذاكرة الرئيسية؛

أ. عدّاد التعليمات
ب. مرصف التعليمات
ج. وحدة التحكّم بالتعليمات
د. وحدة المنطق والحساب

17. أي من مكوّنات المعالج يقوم بتنفيذ التعليمات؟

أ. مرصف التعليمات
ب. وحدة التحكم بالتعليمات
ج. وحدة المنطق والحساب
د. الذاكرة الرئيسية

18. أيٍ من مكونات المعالج يحتفظ بمعلومات التحكم الرئيسية كعنوان التعليمة التالية المُراد تنفيذها؟

أ. المراصف
ب. الذاكرة الرئيسية
ج. وحدة التحكم بالتعليمات
د. وحدة المنطق والحساب

19. يتمّ استحضار تعليمة ما في أثناء ______________.

أ. زمن التعليمة
ب. زمن التنفيذ
ج. زمن الدورة
د. زمن التشغيل

20. يتمّ تنفيذ تعليمة ما في أثناء ______________.

أ. زمن التنفيذ
ب. زمن الدورة
ج. زمن التشغيل
د. زمن التعليمة

21. يتمّ استحضار وتنفيذ تعليمة واحدة في أثناء ______________ واحد(ة).

أ. نبضة ساعة
ب. التعليمة
ج. نيل الذاكرة
د. دورة الآلة

الاجابات

1.ب 2.ب 3.ب 4.ج 5.أ 6.أ 7.أ 8.ج 9.أ 10.ب 11.ب 12.ج 13.د
14.د 15.د 16.ج 17.ج 18.أ 19.أ 20.أ 21.د.

ربط المفاهيم

1. تخزّن المعطيات داخل الكمبيوتر وتعالج في صيغة ثنائية؟ لماذا تعالج ثنائياً؟
2. ما هي أوجه الاختلاف بين الخوينات والخانات والكلمات؟
3. كيف يتمّ نيل الذاكرة الرئيسية في كمبيوتر ما؟
4. ميّز بين عنوان موقع في ذاكرة ومحتوياته.
5. ميّز بين ذاكرة القراءة فقط وذاكرة النيل العشوائي.
6. ما هي التعليمة؟

7. ماذا يطرأ أثناء دورة آلية واحدة؟ وضّح مبدأ الدورة الآلية بالنسبة الى مكوّنات الكمبيوتر الداخلية. بمعنى آخر، اشرح كيفية عمل هذه المكونات من أجل تنفيذ تعليمة ما.

8. ما هي المراصف وما هي أغراض استخداماتها؟

9. في الفصل 1، التمرين 5، كان قد طُلب منك رسم مجمل مكوّنات نظام كمبيوتر نموذجي. وقد درست في هذا الفصل المعالج والذاكرة الرئيسية بشكل أكثر تفصيلاً، أضف ذلك التفصيل الى الرسم التوضيحي.

3.

الدخل والخرج

مفاهيم أساسية

نيل الكمبيوتر

الدخل/الخرج الأساسيان
- ☐ لوحات المفاتيح وشاشات العرض
- ☐ الطابعات

التخطيطات
- ☐ الخرج التخطيطي
- ☐ الدخل التخطيطي

أجهزة الدخل والخرج الأخرى
- ☐ البطاقات المثقبة
- ☐ الطابعات
- ☐ الوسائط المغنطيسية
- ☐ الوسائط البصرية
- ☐ الطرفيات
- ☐ ادراك الصوت والاستجابة له

وصل المكوّنات
- ☐ البينيّات
- ☐ القنوات ووحدات التحكّم

عملية نيل الكمبيوتر

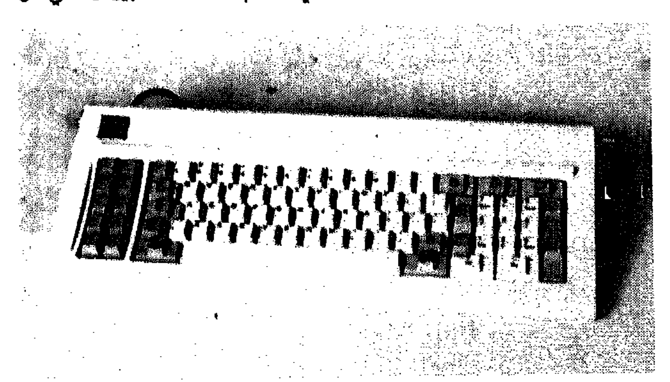
الكمبيوتر آلة تعالج المعطيات لتحويلها الى معلومات. ليس هنالك من داع لمعالجة المعطيات اذا لم يكن الانسان بحاجة اليها. فبدون معطيات ليس هنالك ما يستدعي المعالجة. إن أجهزة الدخل والخرج تتيح الوسيلة اللازمة للانسان لنيل الكمبيوتر. وسنبحث في هذا الفصل عدداً من أجهزة الدخل والخرج ووسائطها وسنوضح بالأمثلة كيفيّة ربطها عضوياً بالكمبيوتر.

الدخل/الخرج الأساسيان

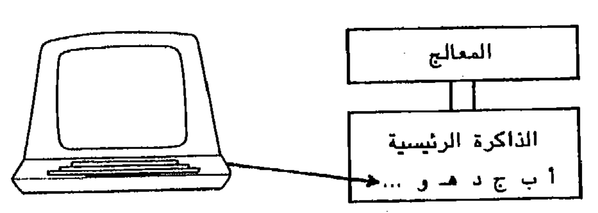
شاشات العرض ولوحات المفاتيح

إن وسيلة الدخل الأساسية المستخدمة في معظم انظمة الكمبيوتر هي **لوحة المفاتيح** (الشكل 3.2). فعند الضرب على مفاتيح الرموز يجري خزنها في الذاكرة الرئيسية (الشكل 3.2أ). فعند الضرب على مفاتيح الرموز يجري خزنها في الذاكرة الرئيسية (الشكل 3.2أ) ومن ثم تنقل من الذاكرة الى جهاز خرج أساسي **كشاشة العرض** (الشكل 3.2ب). والشاشة (الشكل 3.3) التي تُسمى **بالمرقاب**، تستخدم كنافذة على الذاكرة الرئيسية فيتمكّن الشخص الذي يستعملها من رؤية محتوياتها.

الشكل 3.1 وسيلة الدخل الأساسية المستخدمة في معظم انظمة الكمبيوتر هي لوحة المفاتيح.

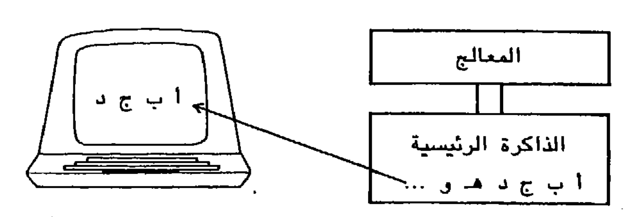


الشكل 3.2 الدخل والخرج



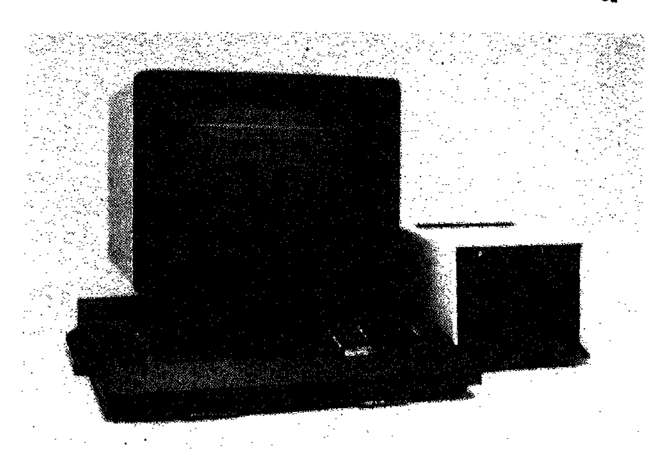
1. بعد أن تطبع الرموز تُخزّن في ذاكرة الكمبيوتر الرئيسية.





ب. يتمّ بعدئذ إخراج رموز مختارة من الذاكرة الرئيسية الى شاشة العرض.

الشكل 3.3 شاشة العرض هي أكثر أجهزة الخرج الأساسي شيوعاً بالنسبة لمعظم أنظمة الكمبيوتر الصغيرة.

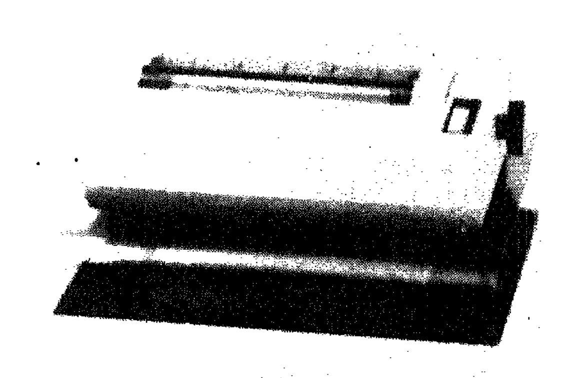


هناك أنواع مختلفة من شاشات العرض، فيظهر بعضها رموزاً بيضاء على خلفية سوداء. وتتضمّن الاختيارات الأخرى شاشات كهرمانية وخضراء اللون. وتظهر المراقيب الملوّنة، الرموز والجداول والصور والرسوم التوضيحية بالألوان. وفي بعض الأحيان، تستخدم شاشات التلفزيون كأجهزة عرض زهيدة الثمن، لكن بسبب نقص في وضوح الاشارات التلفزيونية عندما تعرض عناصر صغيرة (كالحروف والأرقام)، فإن مراقيب الكمبيوتر الحقيقية تنتج صوراً أكثر وضوحاً.

الطابعات

إن الصورة التي تُعرض على شاشات العرض هي صورة موقتة، تختفي حالما ينقطع التيار الكهربائي. فبنقل الخرج الى **الطابعة** (الشكل 3.4)، يمكن الحصول على نسخة دائمة (تُدعى النسخة المطبوعة) وتكون الطابعات الصفيفية النقطية رموزاً على شكل أنماط نقطية، زهيدة الثمن. ولكن قد نجد صعوبة في قراءة الخرج. وتنتج الطابعات ذات الطباعة الجيّدة صور حروف كاملة الشكل واضحة المعالم. وهناك أساليب فنية غير مألوفة تشكّل صوراً بنفث الحبر أو بحرق (كربنة) سطح الورق اختيارياً.

الشكل 3.4 تولد الطابعة نسخاً مطبوعة عن الخرج يمكن حفظها لفترة أطول.

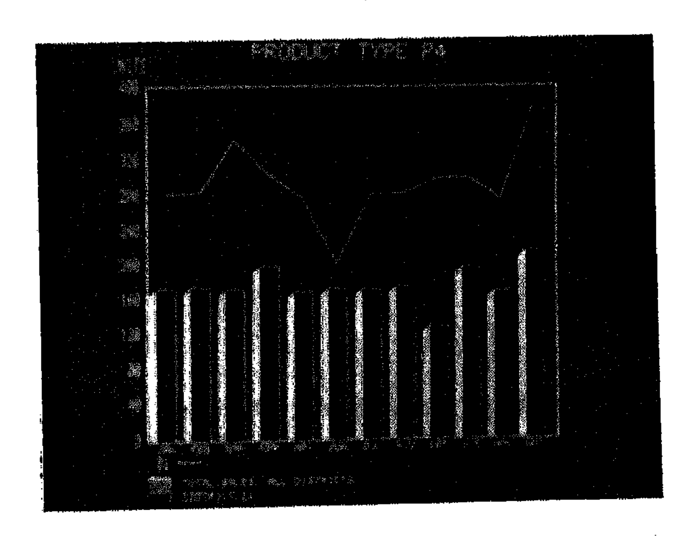


التخطيطات

الخرج التخطيطي

لا تقتصر أجهزة الكمبيوتر على عرض الرموز فحسب بل من الممكن عرض خرج **تخطيطي** (الشكل 3.5) أيضاً. تذكر أن خرج الكمبيوتر يأتي من الذاكرة الرئيسية، وهكذا، فإذا أريد عرض صور تخطيطية وجب أولاً أن ترسم في الذاكرة. وتخزّن الذاكرة الخوينات. كيف يمكن تحديد رمز تمهيدي أو جدول أعمدة أو رسم تخطيطي كنمط خويني؟

الشكل 3.5 تستخدم أجهزة الكمبيوتر بشكل متزايد لانتاج الخرج التخطيطي وخرج الرموز كذلك.

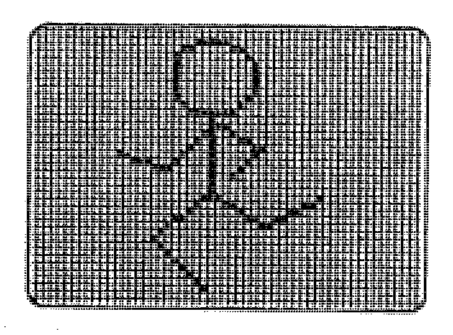


السرّ هو تقسيم الشاشة الى شبكات عناصر صورية أو ما يُسمّى **بالبيكسل** (الشكل 3.6). ويمثّل كل بيكسل نقطة. ويجري تشكيل الصورة بتشغيل النقط والبنوط وايقافها اختيارياً. إن حالة التشغيل والايقاف لكل بيكسل هي في الأساس حالة ثنائية ويمكن خزنها في الذاكرة. ولدى القيام بتفحّص دقيق للذاكرة يجري عرض عناصر الصورة وبالتالي يظهر شكلها.

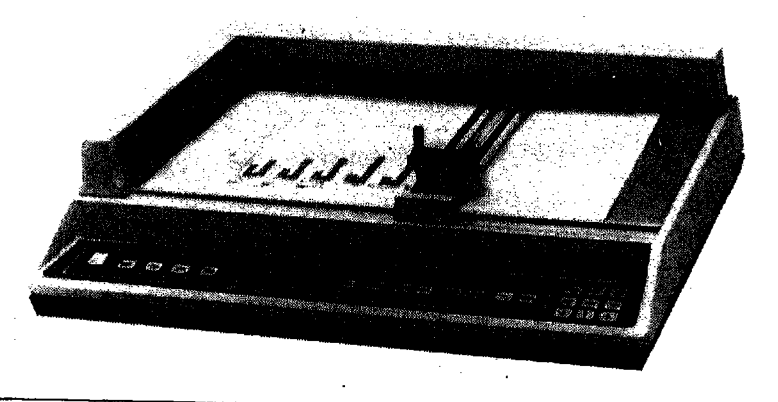
تعتمد نوعية أو **وضوح الصورة** على عمل عدد البيكسلات. فعلى سبيل المثال، قد لا تظهر أية تفاصيل في عنصر صورة كبيرة واحدة، عندئذ تكون الشاشة إمّا كاملة السواد أو البياض. وبوجود شبك يتألف من تسعة بيكسلات (3 × 3) فإنه من الممكن تشكيل رمز واحد تقريبي ويشبه ذلك الى حدّ كبير مصباح الضوء عندما يشكّل رمز إصابة على لوحة تسجيل الأهداف. وبازدياد عدد عناصر الصورة يزداد وضوح التفاصيل المعروضة. وبالطبع، هناك ثمن لهذا التحسّن. وبما انه يجب خزن المعلومات المتعلقة بكل بيكسل، فإن التخطيطات البادية الوضوح تتطلّب ذاكرة تفوق التخطيطات ذات الوضوح المنخفض.

يُعدّ الخرج التخطيطي المعروض على الشاشة مؤقتاً. وبالنسبة للنسخة المطبوعة فانه يمكن ارسال الصور الى راسمة (الشكل 3.7).

الشكل 3.6 للقيام بعرض خرج تخطيطي تُقسم الشاشة الى شبكة مؤلفة من عناصر صورية أو بيكسلات. يمثّل كل بيكسل نقطة (بنط) واحدة وتكّون الصور بتنشيط أو ايقاف كل بيكسل منتقى.



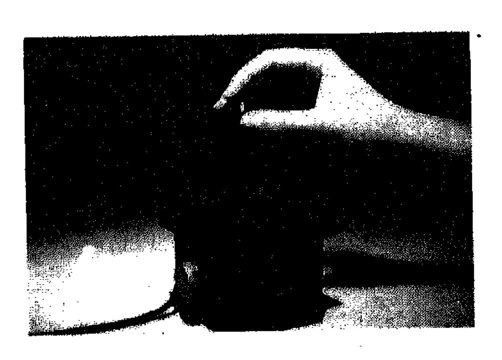
الشكل 3.7 تنتج الراسمة طبعة دائمة من الخرج التخطيطي.



الدخل التخطيطي

كيف يمكن للشخص تداول الصورة والتفاعل معها بعد عرضها مباشرة؟ إن احدى الوسائل لانتاج دخل تخطيطي هي ضبط **الزليقة**. وتظهر الزليقة عادة على شكل خط أو مربع رفاف، يشير الى موضع الرمز المطبوع التالي الذي سيظهر على الشاشة.
ربما تكون المسلاة من أشهر أنواع الأجهزة التي تستخدم للتحكم بالزليقة (الشكل 3.8). وإذا كنت قد زاولت لعبة الكمبيوتر المسلية فمن المرجح انك قد استخدمت جهازاً شبيهاً بها. ويمكن التحكم بالزليقة بواسطة «الفأرة» (الشكل 3.9) وهو جهاز يبلغ حجمه حجم راحة اليد ومزود بدحروجة في قاعها. وتوضع الفأرة على سطح منبسط. اذا حركت الفأرة الى الأمام فستتحرك الزليقة الى فوق وإذا حركتها باتجاه اليسار فستتحرك الزليقة الى اليسار وهكذا دواليك. تتشابه الفأرة مع المسلاة في انها تنقل الزليقة بالنسبة لموضعها الحالي. وتوجد مفاتيح التحكم بالزليقة في العديد من لوحات المفاتيح وتؤدي الوظيفة نفسها.
تحدد الزليقة موضع نقطة واحدة على الشاشة. وبضغط الزر الموجود على المسلاة أو الفأرة أو بكبس مفتاح الادخال على لوحة المفاتيح يتم ادخال الزليقة في وضعها القائم. وبعد أن يحدد مركز الزليقة يمكن للبرنامج المخزون أن يتخذ العمل المناسب. أما بالنسبة للشاشات الملمسية (الشكل 3.10) أو القلم الضوئي (الشكل 3.11) فيقوم المستخدم بإدخال النقطة في غاية السهولة وذلك بلمس بقعة على الشاشة.

الشكل 3.8 ربما تكون المسلاة من أشهر أنواع الأجهزة المستخدمة للتحكم بالزليقة.



الشكل 3.9 كما يمكن التحكم بموضع الزليقة بمداولة الفأرة أو بكبس مفاتيح التحكم بالزليقة الموجودة على لوحة المفاتيح.



الشكل 3.10 يمكن استخدام الفأرة أو المسلاة أيضاً في إدخال الاحداثين السيني والصادي لنقاط مختارة. ومن الخيارات الأخرى استعمال شاشة ملمسية.

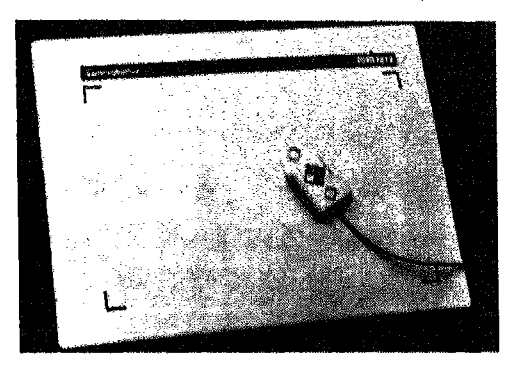


الشكل 3.11 يمكن استخدام قلم ضوئي أو قارئة السمة أيضاً لادخال نقاط مختارة.



يستخدم المرقم (الشكل 3.12) لادخال المعطيات التخطيطية. فتوضَع الوثيقة الأصلية (على سبيل المثال، رسم هندسي) على لوحة ويتمّ وضع نقطة إسناد (تكون في الغالب، عند أسفل الزاوية اليسرى أو في الوسط). ثم يتمّ تحريك المرقم ليبلغ نقطة هامة كنقطة تقاطع سطرين، ويتم الضغط على زر وتنقل نقاط الاحداثين السيني والصادي الى الكمبيوتر. ويستطيع المستخدم أن يدخل جميع النقاط الأساسية عن طريق تحريك المرقم بأسلوب منهجي على الوثيقة الأصلية. وفيما بعد، يمكنُ للراسمة أن تعيد انتاج الرسم بواسطة وصل النقاط.

الشكل 3.12 يمكن ادخال معطيات التخطيطات الى كمبيوتر بواسطة المرقم. والفكرة هي أن نقوم بادخال النقاط الأساسية المتناسبة مع نقطة اسناد ثابتة. وفي وقت لاحق، يمكن للراسمة أن تعيد إنتاج الرسم بواسطة وصل النقاط.

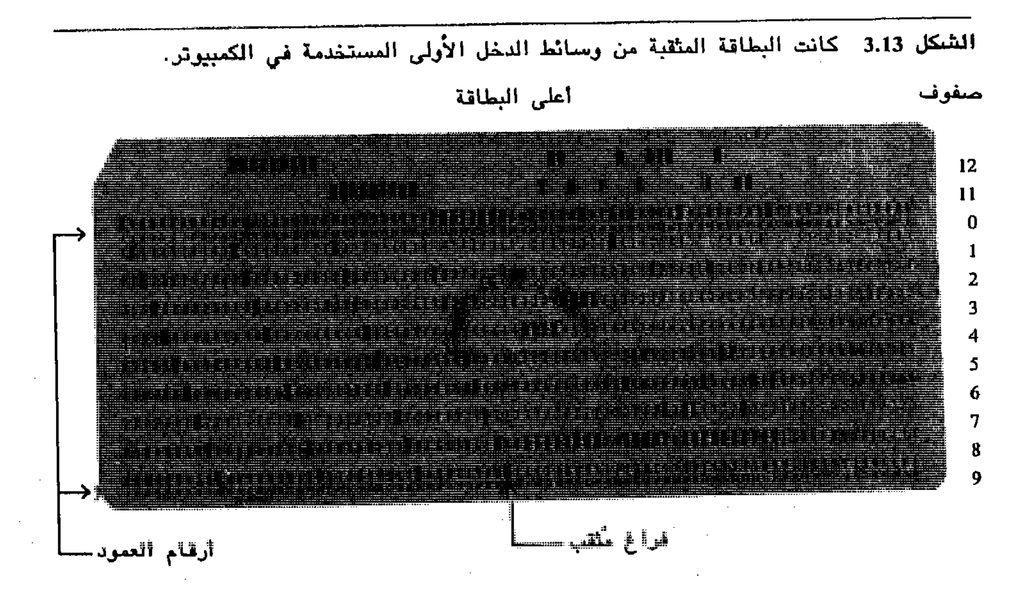


أجهزة أخرى للدخل والخرج

البطاقات المثقبة

كانت البطاقات المثقبة (الشكل 3.13) من وسائط الدخل الأولى المستخدمة في الكمبيوتر. وتتألف البطاقة القياسية من ثمانين عموداً ينقسم كل منها الى اثنَي عشر صفاً. وعندما تطبع السمات عبر مفتاح تثقيب، تسجل على شكل أنماط ثقوب في عمود بحيث قد يكون لكل موقع عمود/صف ثقب أو لا يكون. ويقوم قارئ البطاقات بتحويل أنماط الثقوب الى شكل الكتروني. وتجدر ملاحظة التشابه بين أنماط البطاقات المثقبة/غير المثقبة وأنماط البطاقات الثنائية.

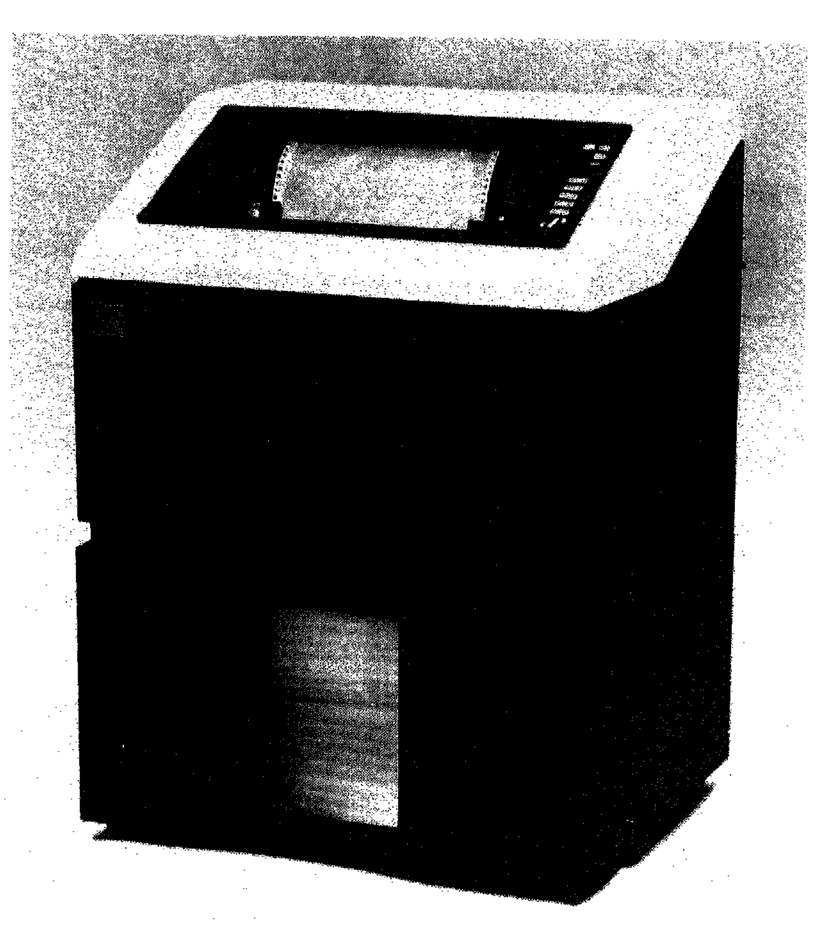
الشكل 3.13 كانت البطاقة المثقبة من وسائط الدخل الأولى المستخدمة في الكمبيوتر.



الطابعات

تطبع معظم طابعات الميكروكمبيوتر سمة واحدة في كل مرة، وذلك بمعدّلات تتراوح عادة، بين 30 الى 180 سمة في الثانية تقريباً. وتُعتبر هذه السرعة جيدة لخرج بضع صفحات مطبوعة. ولكن، لنتصور أننا نطبع تقرير حسابات يتألف من 200 صفحة بسرعة 100 سمة في الثانية. ولنفترض أننا نطبع 120 سمة في كل سطر و50 سطراً في الصفحة، فسيستغرق طبع التقرير ما يفوق الثلاث ساعات. والأسلوب المعقول هنا أن نستخدم طابعة سطرية (الشكل 3.14) تطبع سطراً سطراً بدلاً من سمة سمة كما يدلّ عليه الاسم. وان معدل طباعة 1000 سطر بالدقيقة (أو أكثر) هو أمر عادي، وبهذه السرعة يمكن أن يستغرق طبع تقرير الحسابات الذي ورد ذكره عشر دقائق. كما انه يمكن الحصول على سرعة طبع أكبر باستخدام طابعة صفحات تطبع صفحات كاملة مرة واحدة. وللحصول على المزيد من الخرج المدمج وعلى خزن ذي مدى طويل يمكن استخدام ميكروفيلم الخرج (COM) للكمبيوتر.

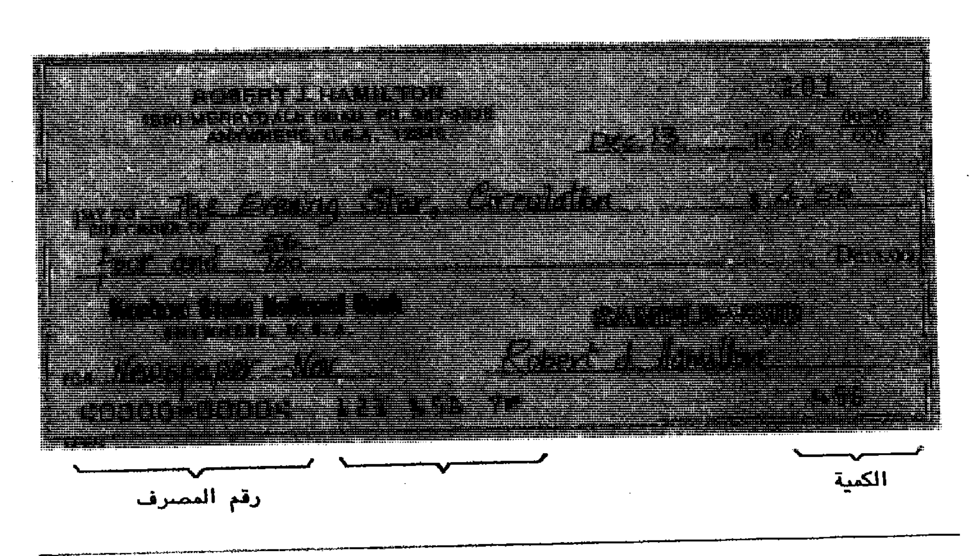
الشكل 3.14 تطبع الطابعة السطرية سطراً كاملاً مرة واحدة.



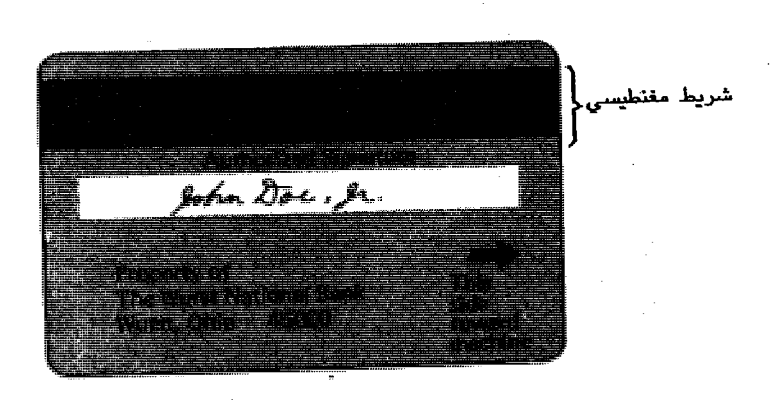
الوسائط المغنطيسية

تعتمد عدّة وسائط دخل شائعة الاستخدام على خصائص مغنطيسية. على سبيل المثال، إن السمات في أسفل معظم الشيكات (الشكل 3.15) مطبوعة بحبر مغنطيسي خاص يُدعى (MICR) أي (كشف الرمز الحبري الممغنط). ويمكن قراءة هذه السمات الكترونياً. كما ان هناك وسيطاً مغنطيسياً آخر يستخدم في المصارف الا وهو البطاقة الشريطية المغنطيسية (الشكل 3.16). إن قطعة الشريط الممغنط تحتفظ بمعطيات معيّنة كرقم حساب الزبون وحدود رصيده وتشبه قراءتها الى حدّ كبير قراءة شريط مسجل للصوت.

الشكل 3.15 يتمّ طبع معظم الشيكات المصرفية باستخدام حبر مغنطيسي خاص يُسمّى «كشف الرمزالحبري الممغنط» (MICR).



الشكل 3.16 بطاقة شريطية مغنطيسية.



الوسائط البصرية

تتمّ قراءة الوسائط الأخرى **بصرياً**. على سبيل المثال، لنفكّر ملياً في أوراق الاختبار القياسية (الشكل 3.17). يستخدم الطلاب في هذه الأوراق قلماً أسود ليعلم الأجوبة التي ينتقيها والورقة البيضاء تعكس الضوء فيما تعكس البقع السوداء الضوء بنسبة أقل. ويمكن تحويل التغييرات في كثافة انعكاس الضوء الى نمط الكتروني. وتستخدم معدات التمييز البصري للسمات (OCR) المبدأ نفسه لقراءة المواد المطبوعة أو حتى المكتوبة باليد. ويمكن قراءة الكودات القضيبية مثل الكود العالمي للمنتجات (UPC) المطبوعة على معظم الطرود في المخازن الكبرى والتي يمكن قراءتها في مركز تدقيق البضائع الخارجة (الشكل 3.18) أو بواسطة مجموعة متنوعة من الماسحات التي تمسك باليد (الشكل 3.19).

الشكل 3.17 تتمّ قراءة صحيفة النقاط التي تمّ احرازها في الامتحان بصرياً.

TESTING FORM

ACADEMIC COMPUTER SERVICE
MIAMI UNIVERSITY
OXFORD, OHIO

SOCIAL SECURITY NUMBER

NAME ______ LAST FIRST MIDDLE

COURSE ______ SECTION ______

INSTRUCTOR ______

INSTRUCTIONS
- MAKE DARK MARKS THAT FILL CIRCLE COMPLETELY
- ERASE CLEANLY ANY ANSWER YOU WISH TO CHANGE
- MAKE NO STRAY MARKS ON EITHER SIDE OF THIS FORM

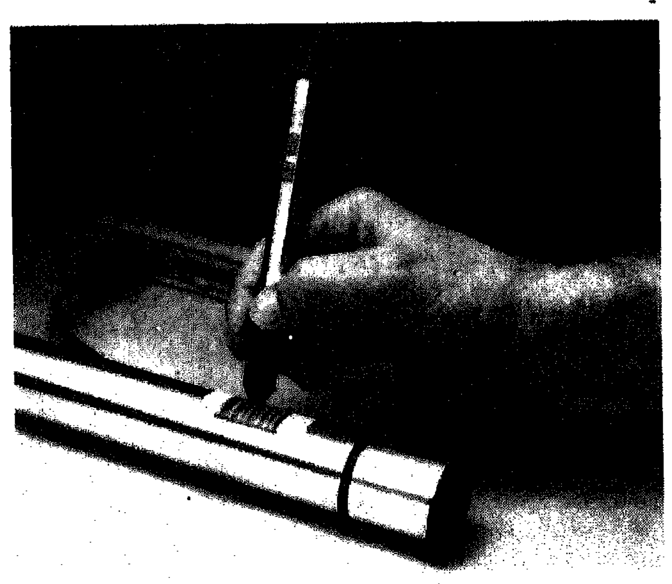
SCORE

FORM NUMBER

الشكل 3.18 تحتوي نقاط تدقيق البضائع الخارجة في المخازن الكبرى على تجهيزات ماسحات وظيفتها قراءة الكودات العالمية للمنتجات المطبوعة على معظم البضائع.



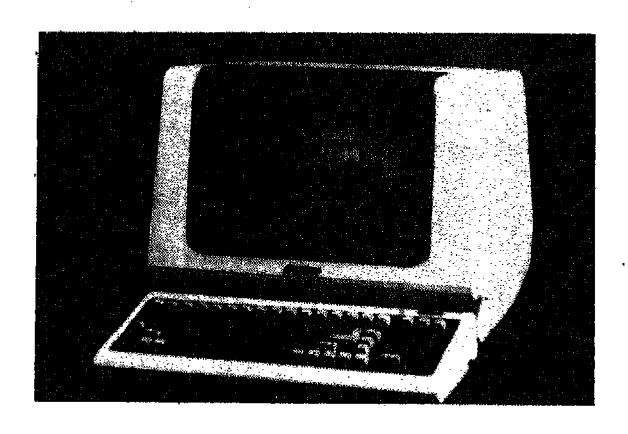
الشكل 3.19 كما تستطيع الماسحات التي تمسك باليد ان تقرأ الكود العالمي للمنتجات ويستخدمها في الغالب، موظفو المبيعات لجمع معطيات عن جرد الموجودات من البضائع والمبيعات أيضاً.



الطرفيات

تُعتبر **الطرفيات** أيضاً (الشكل 3.20) رائجة جداً. وغالباً ما تكون مئات منها أو أكثر موصولة الى كمبيوتر مركزي واحد بواسطة خطوط اتصال. وما الطرفية «المحدودة» الوظيفة سوى لوحة مفاتيح وشاشة عرض. وتحتوي الطرفية الذكية على ذاكرة ومعالج خاصين بها وباستطاعتها أن تنجز عدة وظائف معالجة معطيات بمفردها. أمّا الطرفيات الأخرى المخصّصة لغرض معين فهي مصممة لتقوم بوظيفة معينة. ونجد أمثلة عنها في طرفيات المعاملات المصرفية الأوتوماتية (الشكل 3.21) وطرفيات نقاط التدقيق في البضائع الخارجة لمخزن كبير والتي ورد وصفها سابقاً.

الشكل 3.20 الطرفية عبارة عن وحدة تتألف من لوحة مفاتيح/شاشة عرض موصولة الى كمبيوتر مركزي عبر نوع من خطوط الاتصال.



الشكل 3.21 تشكّل آلة المعاملات المصرفية الأوتوماتية نموذجاً جيداً لطرفية ذات غاية خاصة.



الشكل 3.22 إن جهاز التعرّف على الصوت اصبح الآن متوفراً لبعض الاستعمالات المحدودة.



ادراك الصوت والاستجابة له

قد يكون الصوت أكثر الوسائل الطبيعية ملاءمة للاتصال بالكمبيوتر. والاستجابة للصوت (الخرج) أمر يستخدم بالفعل في تطبيقات ملموسة وسهلة مثل العاب الأطفال والفيديو. ونظراً للتنوّع الكبير في انماط الكلام البشري يعتبر التعرّف على الصوت (الدخل) أكثر صعوبة من الاستجابة له إلا انه تمّ تحقيق تقدم مهم في هذا الصدد. إن التعرف على الصوت متوفر هنا بالفعل لبعض التطبيقات المحدودة (الشكل 3.22).

وصل المكونات

البينيات

نعلم ان المعطيات تُخزّن في الكمبيوتر في أنماط خوينات وتكون هذه الأنماط منسجمة بعضها مع بعض ضمن آلة معينة، على سبيل المثال، إذا كان الكود للحرف A على نمط 01000001 فإن هذا النمط هو الذي سيستخدم فقط لتمثيل الحرف A داخل الكمبيوتر.

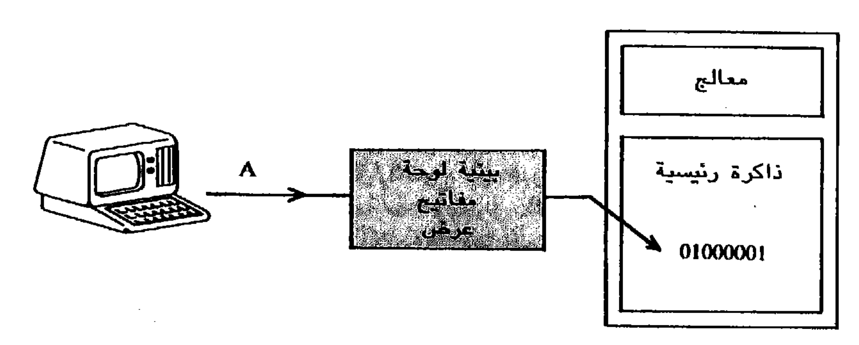
إن هذه القاعدة لا تُطبّق على أجهزة الدخل أو الخرج. فعلى لوحة المفاتيح، يولد كل مفتاح سمة واحدة. وفي التخطيطيات يتمّ عرض بيكسلات. أمّا الطابعة الصفيفية النقطية فتقوم بتمثيل السمات كأنماط نقطية وتقوم قارئة البطاقات بتفسير انماط الثقوب التي أحدثت في بطاقة. ويقرأ جهاز بصري معدّل قوة الضوء، بينما يقرأ جهاز مغنطيسي معدل القوة المغنطيسية. وكل جهاز دخل أو خرج يمثل أو يفسر المعطيات بطريقته المنفردة الخاصة، وقد تلائم الاشارات المستخدمة من قبل جهاز محيطي الاشارات المخزونة داخل الكمبيوتر أو لا تلائمها. وإذا كان على هذين الجهازين غير المتشابهين أن يتصلا فان الترجمة تصبح ضرورية وهذه هي وظيفة اللوحة البينية.

على سبيل المثال، لنبحث في لوحة المفاتيح، فعندما يجري ضغط أحد المفاتيح يتمّ ارسال اشارة الكترونية الى الموصل البيني للوحة المفاتيح ورداً على ذلك، يولد الموصل البيني الكود الذي يستخدم لتمثيل تلك السمة داخل الكمبيوتر وينقل المعطيات التي نظمت

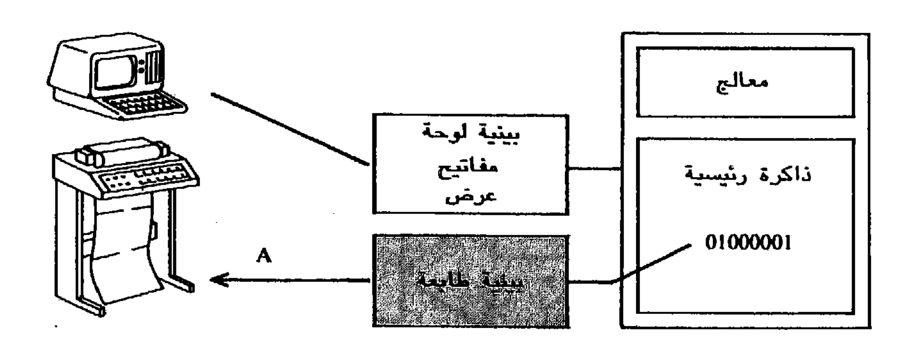
بشكل كود معين الى الذاكرة الرئيسية (الشكل 3.23أ). دعنا نغير الجهاز الى طابعة (الشكل 3.23 ب) فعند بدء الخرج تخزن المعطيات داخل الكمبيوتر على صورة سمات مكودة ثنائياً. وتتطلب الطابعة نمطا نقطيا لذلك. وعندئذ يتضح لنا أن الترجمة ضرورية. وهكذا ترسل السمات المكودة الى بينية الطابعة التي تقوم بترجمة كودات الكمبيوتر الثنائية الى شكل يصدر عن الطابعة.

تختلف الطابعة عن لوحة المفاتيح، فالاشارات التي تتحكم مادياً بكل منهما والأنماط الالكترونية التي تستخدمها لتمثيل المعطيات تعتمد على الجهاز. مع ذلك، ولأن المهام المعتمدة على الجهاز يتم تعيينها الى لوحات الموصلات البينية المختصة يمكن وصل كليهما معاً الى الكمبيوتر نفسه. وعند الدخل، يقوم الموصل البيني للوحة بترجمة الاشارات الخارجية الى شكل مقبول لدى الكمبيوتر. ويتم تحويل اشارات الخرج الكترونيا من الكود الداخلي للكمبيوتر الى شكل مقبول لدى الجهاز المحيطي. ونظراً لاختلافهما الكترونيا تتطلب الطابعة ولوحة المفاتيح لوحات بينية مختلفة. وفي الواقع، يحتاج كل جهاز دخل أو خرج الى لوحة بينية منفردة لترجمة اشاراته المعتمدة على الجهاز الى الكود الداخلي للكمبيوتر أو منه.

الشكل 3.23 وظائف اللوحة البينية.



أ. تدخل معطيات الدخل من لوحة المفاتيح الى البينية ثم يتم تحويلها الى شكل الكمبيوتر الداخلي.



ب. ترسل المعطيات المخزونة في الذاكرة الرئيسية الى بينية الطابعة ثم تُحوّل الى نمط طابعة ويتم إخراجها.

تحتوي لوحات بينيات عديدة على **وسائط** خزن مرحلية. والمخزن الوسيط المرحلي هو بمثابة وسيط خزن مؤقت يُستخدم لتعديل فارق السرعة بين جهازين. على سبيل المثال، إذا سبق وانتظرت حتى يتم طبع ورقة مطوّلة فأنت لا شك مدرك أن الطابعة أكثر بطءاً من الكمبيوتر. وإذا كان هذا الإنتظار يسبّب لك مشكلة فما عليك إلاّ أن تضيف مخزناً وسيطاً مرحلياً على الموصل البيني للطابعة، بهذا وبدلاً من أن يرسل الكمبيوتر محتويات الذاكرة الرئيسية مباشرة الى الطابعة، يمكنه أن يرسل المعلومات الى المخزن الوسيط المرحلي بالسرعة التي يعمل بها الكمبيوتر. وبعد مرور وقت قصير يكون قد تمّ خلاله تفريغ السمات من المخزن الوسيط المرحلي الى الطابعة بالسرعة التي تعمل بها الطابعة، يمكنك أن تستخدم الكمبيوتر للقيام بمهمة أخرى.

القنوات ووحدات التحكّم

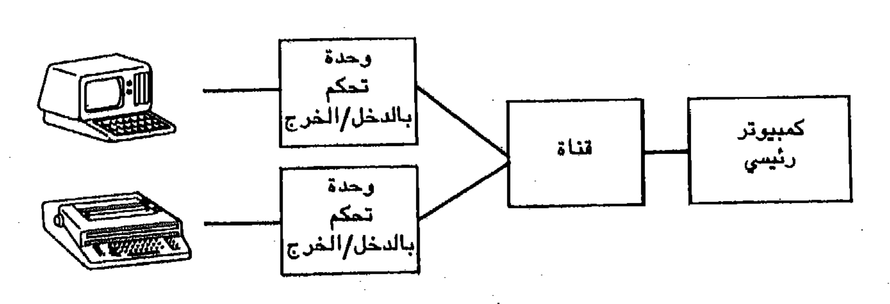
إن استخدام لوحة بينيّة لكل جهاز في نظام الميكروكمبيوتر هو أمر معقول لكنه ليس كذلك بالنسبة لنظام واسع النطاق له مئات الأجهزة المحيطية. وبدلاً من ذلك، يتمّ ربط أجهزة الدخل والخرج الى نظام كمبيوتر واسع النطاق عبر **قنوات ووحدات تحكم**.

إن بعض الوظائف شائعة الاستخدام بالنسبة لجميع أنواع الدخل والخرج تقريباً (من أمثال هذه الوظائف تحديد الموضع الذي توجد أو تخزّن فيه الخانة القادمة في الذاكرة، وعد السمات المنقولة الى جهاز خارجي أو منه. وفي الميكروكمبيوتر، يتمّ إنجاز هذه الوظائف بواسطة كل لوحة بينية وبالفعل يتم مضاعفتها لكل جهاز في النظام. أمّا في الآلات الأكبر حجماً، فيتمّ تعيين هذه الوظائف العادية الى قنوات المعطيات (الشكل 3.24).

يجب أن نلاحظ أن القناة تشرف على تدبير الوظائف المستقلة عن الجهاز ولكن ماذا بشأن الوظائف المعتمدة على الجهاز مثل تفسير الأنماط المغنطيسية؟ يتمّ تنفيذ هذه الوظائف عبر وحدات التحكم بالدخل/الخرج أو عبر وحدات البينيات (الشكل 3.24). لكل جهاز مادي وحدة تحكم خاصة به. وتجري القناة الاتصال مع الكمبيوتر بلغة الكمبيوتر كما تجري وحدة التحكّم الاتصال مع الجهاز الخارجي باللغة التي يفهمها هذا الجهاز. وتعمل القناة ووحدة التحكّم معاً، فيقومان بالترجمة.

قد يكون لنظام كمبيوتر كبير الحجم ونموذجي ثلاث أو أربع قنوات تتصل بكل منها وحدات تحكم عديدة. ويُعتبر هذا الأسلوب مرناً للغاية إذ انه يسمح لمئات من أجهزة الدخل والخرج بالوصول الى الكمبيوتر عبر بضعة ممرّات معطيات يسهل التحكّم بها.

الشكل 3.24 في نظام كمبيوتر كبير الحجم، يتمّ ربط الأجهزة المحيطية الى النظام عبر قناة ووحدة تحكم بالدخل/الخرج.



الخلاصة

يتمكّن الانسان من نيل الكمبيوتر عبر أجهزة الدخل والخرج الخاصة به. وجهاز الدخل الأساسي الذي يستخدم في معظم أنظمة الكمبيوتر ذات الحجم الصغير هو لوحة المفاتيح. فعندما يقوم المستخدم بطبع سمات، يتم خزنها في الذاكرة الرئيسية للكمبيوتر ثم يتم خرجها الى شاشة العرض (جهاز الخرج الأساسي) حيث يتمكن المستخدم من رؤيتها.

إذا دعت الحاجة للحصول على نسخة دائمة عن معطيات الخرج، يمكن إرسال النسخة من الذاكرة الى الطابعة. إن الكمبيوترات ليست محصورة بخرج السمات، ومع ذلك هناك كثير من الأنظمة التي تدعم التخطيطيات. وتنقسم شاشة العرض الى عدد من النقاط تُسمى بيكسلات ويتمّ تشكيل الصور عن طريق تنشيط وايقاف البيكسلات حسب الاختيار. أمّا درجة وضوح الشاشة فهي دالة لعدد البيكسلات. ولقد بحثنا بشكل موجز، عدة أجهزة ووسائط دخل وخرج أخرى من ضمنها البطاقات المثقبة ونماذج متنوعة من الطابعات والوسائط المغنطيسية والبصرية والطرفيات والدخل/الخرج الصوتي.

في النهاية، اتجهنا نحو مسألة ربط أجهزة الدخل والخرج بنظام الكمبيوتر. ووجدنا أن كل جهاز يختلف الكترونياً عن الآخر. إلاّ ان الكمبيوتر يتعامل دوماً مع كود مشترك، هنا تعمل اللوحة البينية على إجراء الاتصال لسدّ هذه الثغرة. يرسل جهاز دخل معطياته الى البينية التي تقوم بتحويل هذه المعطيات من نمط الجهاز الخارجي الى نمط الكمبيوتر الداخلي ويخزّنها في الذاكرة الرئيسية. وعند الخرج، تتحرّك المعلومات من الذاكرة الى البينية حيث يتم تحويلها الى نمط الجهاز الخارجي ويتم خرجها. ويمكن أن يساعد الوسيط المرحلي في تعديل فارق السرعة بين جهازين متجاورين، وفي الكمبيوترات ذات الأحجام الكبيرة، يتم ربط كل جهاز خارجي بوحدة تحكم. وتكون وحدات التحكم موصولة بقنوات تكون بدورها موصولة الى الكمبيوتر.

لقد ركّزنا في هذا الفصل على الدخل والخرج، وأوضحنا كيف يمكن أن يجري الانسان الاتصال مع الكمبيوتر. ومن إحدى فوائد ادخال المعطيات في الكمبيوتر هو أنه متى تمّ تحويلها الى نمط الكتروني يمكن استخدامها مراراً وتكراراً. وفي الفصل التالي سنأخذ بالاعتبار الخزن المساعد الذي تطول مدته.

مصطلحات أساسية

- ☐ وسيط (مرحلي)
- ☐ قناة
- ☐ وحدة تحكم بالدخل/الخرج
- ☐ زليقة
- ☐ شاشة عرض
- ☐ تخطيطيات
- ☐ بينية
- ☐ لوحة مفاتيح
- ☐ وسائط مغنطيسية
- ☐ مرقاب
- ☐ وسائط بصرية
- ☐ بيكسل
- ☐ طابعة
- ☐ بطاقة مثقبة
- ☐ وضوح
- ☐ طرفية
- ☐ دخل/خرج صوتي

اختبار ذاتي

1. إن جهاز الدخل الأساسي في كمبيوتر صغير الحجم هو ________.

أ. شاشة العرض ج. لوحة المفاتيح
ب. الطابعة د. قارئة البطاقات

2. إن جهاز الخرج الأساسي في كمبيوتر صغير الحجم هو ________.

أ. لوحة المفاتيح
ب. شاشة العرض
ج. قريص مرن
د. راسمة

3. تولد ________ نسخة مطبوعة عن معطيات الخرج.

أ. وحدة العرض
ب. الشاشة
ج. المرقاب
د. الطابعة

4. تنقسم شاشة العرض الى شبكة من ________.

أ. الخانات
ب. الخوينات
ج. البيكسلات
د. الكلمات

5. تشير أو يشير ________ الى مكان ظهور السمة التالية على الشاشة.

أ. البيكسل
ب. عنصر الصورة
ج. الزليقة
د. الاحداثات السينية والصادية

6. على ________ تسجل المعطيات بأنماط ثقوب.

أ. طابعة صفيفية نقطية
ب. بطاقة مثقبة
ج. مرقم
د. جميع الأشياء المذكورة في أ، ب، ج

7. تتم قراءة السمات في أسفل الشيك ________.

أ. بصرياً
ب. بالملامسة
ج. مغنطيسياً
د. لا تتم قراءتها

8. تتم قراءة أوراق الاختبار القياسية ________

أ. مغنطيسياً
ب. بالملامسة المادية
ج. بصرياً
د. بواسطة الطرفيات

9. في الغالب، يتم ربط مئة أو أكثر من ________ بكمبيوتر مركزي أحادي بواسطة خطوط اتصال.

أ. أجهزة التمييز البصري للسمات
ب. الطرفيات
ج. الماسحات
د. الطابعات

10. يتم ربط الأجهزة الخارجية بنظام كمبيوتر صغير الحجم عبر ________.

أ. قنوات
ب. وحدات تحكم
ج. قوابس كهربائية
د. بينيات

11. يقوم أو تقوم ________ بعملية ترجمة ما بين الكودات الداخلية للكمبيوتر والكودات الخارجية للأجهزة المحيطية.

أ. المعالج ج. الذاكرة
ب. البينية د. المرصف

12. ______ هي وسيط خزن موقت يستخدم لتعديل التفاوت في السرعة بين جهازين متجاورين.

أ. مخزن وسيط مرحلي ج. وحدة التحكم
ب. البينية د. القناة

13. في نظام كمبيوتر كبير الحجم، يربط أو تربط ______ الأجهزة الخارجية بالكمبيوتر كما وينجز أو تنجز عدداً من المهام المستقلة عن الجهاز

أ. القناة ج. وسيط خزن مرحلي
ب. وحدة التحكم د. المرصف

14. في نظام كمبيوتر كبير الحجم، يربط أو تربط ______ الأجهزة الخارجية بقناة.

أ. وحدة التحكم ج. المرصف
ب. وسيط خزن مرحلي د. يكون الربط مباشراً

الاجابات

1.ج 2.ب 3.د 4.ج 5.ج 6.ب 7.ج 8.ج 9.ب 10.د 11.ب 12.أ 13.أ 14.أ.

ربط المفاهيم

1. ما هي وظائف أجهزة الدخل والخرج للكمبيوتر؟
2. ضع قائمة بعدة أجهزة دخل وعدة أجهزة خرج. كيف يمكنك أن تفرّق بينها. لا تكرّر في إجابتك ما جاء في الكتاب. بل فكر في إجابتك على السوأل.
3. ما هي البيكسل؟ قم بربط البيكسلات مع وضوح الشاشة.
4. ما هي الزليقة؟ ضع قائمة بأسماء عدة أجهزة للتحكم بموضع الزليقة.
5. إشرح بإيجاز كيفية عمل الوسائط المغنطيسية.
6. اشرح بإيجاز كيفية عمل الوسائط البصرية.
7. ما هي البينية؟ ولماذا تعتبر البينيات ضرورية؟
8. ما هو وسيط الخزن المرحلي؟ ولماذا تعتبره ضرورياً؟
9. ما هي القناة؟ وما هي وحدة التحكم بالدخل/الخرج؟ ميّز بين الوظائف التي تنجزها القناة والوظائف التي تنجزها وحدة التحكم بالدخل/الخرج.
10. في الميكروكمبيوترات، يتم ربط الأجهزة المحيطية بالنظام عبر لوحات بينية. وفي الكمبيوترات الرئيسية الكبيرة، يتم ربط الأجهزة المحيطية عبر قنوات ووحدات تحكم بالدخل/الخرج. لماذا؟

4.

الخزن المساعد

مفاهيم أساسية

لماذا تستخدم أجهزة الخزن المساعد؟

وسائط الخزن المساعد

- ☐ كاسيت مغنطيسي
- ☐ قريص مرن
- ☐ قرص صلب
- ☐ وسائط أخرى للخزن المساعد

نيل الخزن المساعد

لماذا تستخدم أجهزة الخزن المساعد؟

إحدى حسنات استخدام الكمبيوتر هي في أنه متى تمّ إدخال المعطيات أمكن خزنها على الآلة ونيلها مرات عديدة. والمثال الجيد على ذلك إعداد بطاقات عناوين المشتركين في مجلة، فبدلاً من إعادة طبع جميع هذه البطاقات لكل عدد من المجلة يتمّ إدخال المعطيات الخاصة بكل مشترك مرة واحدة وخزنها ثم تفريغها من المخزن عند الحاجة. وتُعتبر البرامج مثلاً آخر فهي، كمعطيات المشتركين في المجلة، تُخزّن على الكمبيوتر ويتمّ نيلها عند الطلب.

أين يتمّ خزن المعطيات والبرامج بالضبط؟ الاجابة الواضحة على ذلك هي الذاكرة الرئيسية، لكن هذه الذاكرة غالية الثمن وتجهيز معظم الآلات بها محدود. وهناك مشكلة أخرى تتمثّل في عدم استقرار الذاكرة؛ فالذاكرة الرئيسية تفقد محتوياتها عند انقطاع التيار الكهربائي. وبالتالي فنحن بحاجة الى امتداد للذاكرة الرئيسية يتميّز بسرعة الأداء والدقّة والاستقرار زهيد الثمن وذي سعة عالية. وأجهزة **الخزن المساعد** تفي بهذه الحاجة.

وسائط الخزن المساعد

كاسيت مغنطيسي

إن **الكاسيت** المغنطيسي هو من وسائط الخزن المساعد الأقل كلفة، وهو كشريط الكاسيت نفسه الذي يُستخدم لتسجيل الموسيقى. وتتمّ عملية التسجيل بخرج المعطيات الى المسجّل الشريطي، فعند إعادة تشغيل «التسجيل» تُعاد المادة الى الذاكرة الرئيسية. والكاسيتات زهيدة الثمن ومدمجة، ولكنها أيضاً بطيئة الخزن نسبياً ومعرّضة للخطأ، وهي تستخدم في بعض أنظمة الكمبيوتر المنزلية الصغيرة الحجم أو للخزن «الأرشيفي».

قريص مرن

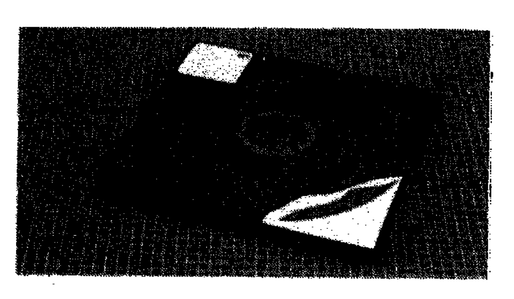
يُعَد **القريص** أو القرص المرن (شكل 4.1) من وسائط الخزن المساعد الأكثر استخداماً في الميكروكمبيوترات. وهو قطعة رقيقة مستديرة من مادة البوليستر المرنة تغلّفها مادة مغنطيسية. وتسجّل المعطيات على أحد سطحَي القريص المستويين او على كليهما معاً. وبما أن ملامسة الغبار أو النسالة أو حتى ملامسة أصابع اليد البشرية للقريص يمكن أن تفسد المعطيات، فإن لكل قريص مرن غلاف وقاية خاصاً به. ويعمل مدوار القريص المرن بصورة مشابهة جداً لقرص التسجيل الدوّار. ويسمح الثقب المستدير في وسط **القرص** للمدوار لآلية التدوير بأن تعشق وتديرها ثم تقوم **آلية النيل** التي تشابه قوتها قوة الذراع بالقراءة من، والكتابة على، سطح القريص، عبر نافذة مرئية قريبة من أسفل الشكل 4.1.

وتُسجِّل المعطيات على سلسلة من الدوائر المتحدة المركز تُسمّى **سككاً** (شكل 4.2)، وتنتقل آلية النيل من سكة الى اخرى، وتقوم في أثناء ذلك إمّا بالقراءة أو بالكتابة. وتنقسم السكك الى **قطاعات**، تتحرك محتويات كل قطاع منها ما بين القريص المرن والذاكرة الرئيسية. ولتمييز القطاعات تتمّ عنونتها تسلسلياً بأعداد مثل 0، 1، 2 الى آخره.

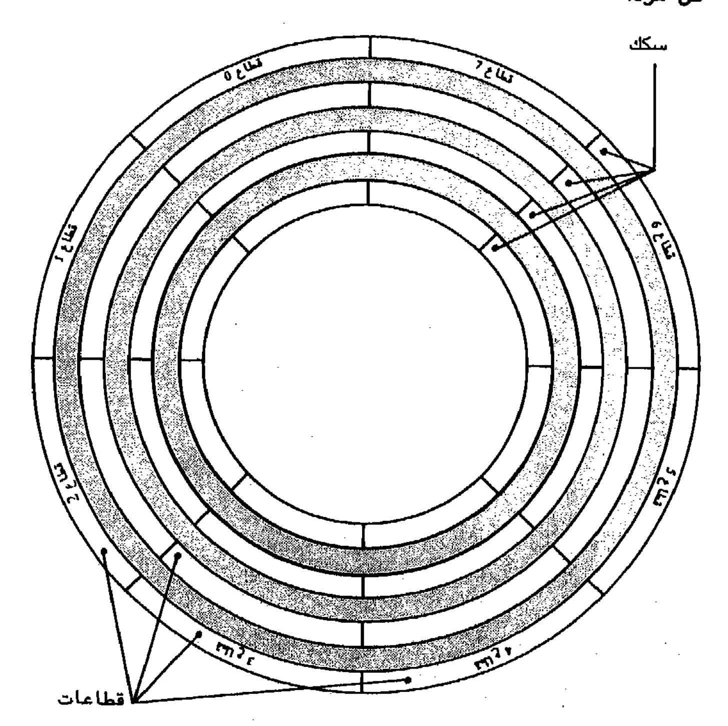
وعندما تصادف تعليمة برنامج تتطلّب ادخال المعطيات من القريص المرن، يُرسل المعالج اشارة تحكّم الى المدوار، وتجاوباً مع ذلك يتعشّق عمود الدوران وثم يبدأ القرص بالدوران ولا يلبث أن يبلغ سرعة دائرية ثابتة (شكل 4.3أ). ثم تتحرّك آلية

النيل الى السكة التي تحتوي على المعطيات المطلوبة (شكل ب4.3). والفترة الزمنية التي يحتاجها المدوار لكي يبلغ سرعة ثابتة تمكّنه من تحديد موضع النيل تُسمّى **بزمن النشد**. ويجب التذكّر أن المعطيات تنقل بين القريص المرن والذاكرة الرئيسية قطاعاً واحداً في كل مرة. وقد يكون القطاع المطلوب في أي مكان على السكة. والفترة الزمنيةالمطلوبة لدوران القطاع الذي يسمح بالنيل (شكل ج4.3) تُسمّى **بالتأخير الدوراني**.

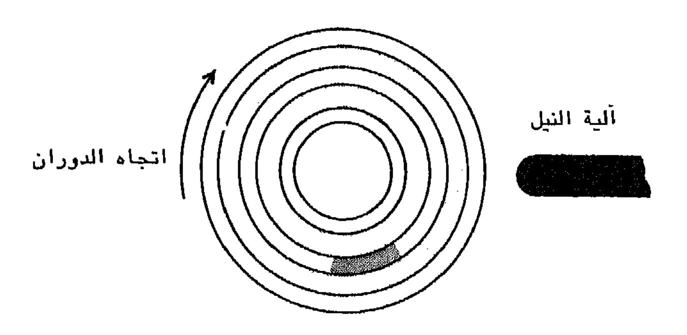
الشكل 4.1 إن القريص المرن هو من وسائط الخزن المساعد الأكثر انتشاراً في الميكروكمبيوترات.



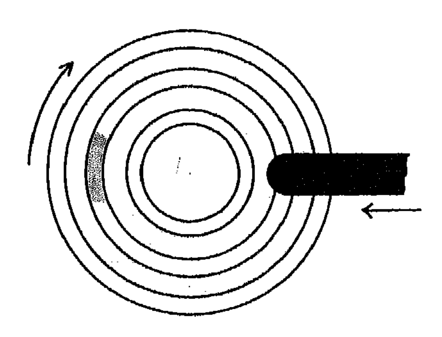
الشكل 4.2 تسجّل المعطيات على سلسلة من الدوار المتحدة المركز تُسمّى سككاً وتنقسم هذه، بدورها، الى قطاعات. تتحرك المعطيات بين سطح القرص والذاكرة الرئيسية قطاعاً واحداً في كل مرة.



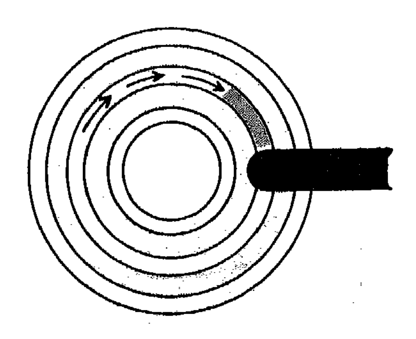
الشكل 4.3 قراءة قطاع من قرص.



أولا. تزداد سرعة مدوار القرص لتبلغ معدل السرعة التشغيلية.



ب. ثانياً: يُحدَّد موقع آلية النيل على السكة التي تخزّن المعطيات المطلوبة. والوقت المطلوب لانجاز الخطوتين أ وب تُسمى النشد.



ج. ثالثاً، يجب أن ينتظر النظام دوران القطاع المطلوب حتى يبلغ راس القراءة والكتابة (التأخير الدوراني) قبل نقل المعطيات الى الذاكرة الرئيسية.

مع ان سرعة الخزن على قريص مرن أكبر بالتأكيد من سرعة الخزن على كاسيت، يظل هناك تأخير في نيل المعطيات يبلغ جزءاً من الثانية على الأقل. وهناك العديد من تطبيقات الكمبيوتر الشخصي الشائعة يشتمل على قرص نيل محدود، لذلك بالكاد يمكن ملاحظة التأخير. ومع ذلك، يمكن أن يكون التأخير في التطبيقات الأخرى غير مقبول والحل يتمثّل في الغالب باستخدام **القرص الصلب**.

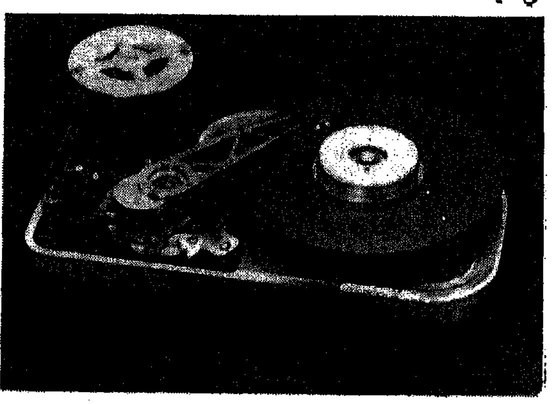
القرص الصلب

يدور مدوار القريص المرن عند قراءة أو كتابة المعطيات فقط. ويجب أن تزداد سرعة المدوار حتى تبلغ معدل سرعة التشغيل قبل أن تتمكن رؤوس القراءة والكتابة من التحرّك ومن نيل المعطيات، وهذا يستغرق وقتاً. وبالمقابل، يفتل القرص الصلب (شكل 4.4) بشكل مستمر وثابت. وبما أن لا ضرورة لانتظار بلوغ المدوار معدل سرعة التشغيل قبل تحريك آلية النيل، يقل زمن النشد بصورة ملموسة، ويبلغ غالباً، بضعة آلاف منالثانية. ويتمّ الحصول على مزيد من التحسينات بدوران القرص على نفسه بسرعة أكبر (بمعدل 1000 دورة في الدقيقة أو أكثر)، الأمر الذي ينقص من معدل التأخير الدوراني. وبالتالي، يمكن نيل المعطيات المخزونة على قرص صلب بسرعة أكبر بكثير من سرعة نيل المعطيات المخزونة على قريص مرن.

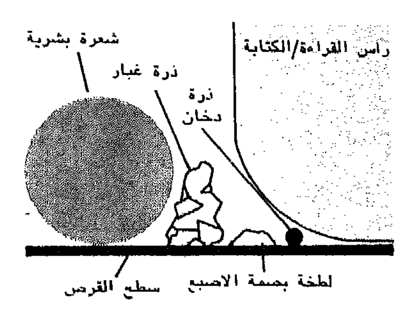
وهناك حسنة أخرى من استخدام القرص الصلب تتمثّل في سعة الخزن. فقد يستطيع قريص مرن نموذجي ذو جانبين أن يخزّن 360,000 سمة، بينما يخزّن قرص صلب في نظام ميكروكمبيوتر من 20 الى 30 مليون سمة.

وفي مداوير القريصات المرنة البطيئة، تركب آلية النيل مباشرة على سطح القرص. وإذا كانت تعمل بسرعة 1000 دورة في الدقيقة، فإن أي تلامس مادي يتم بين سطح القرص ورأس القراءة والكتابة يؤدّي الى تدمير الاثنين معاً. لهذا تركب آلية النيل في القرص الصلب على وسادة من الهواء تعلو سطح القرص بضعة ملايين من البوصة (شكل 4.5) (يكون رأس القراءة والكتابة بشكل سطح انسيابي هوائي يساعده على الارتفاع، أو على ضبط حركته بالاستفادة من تيار الهواء الذي يندفع عبره). ولمنع دخول الملوّثات كذرات الدخان والغبار والشعر بين رأس القراءة والكتابة وسطح القرص، يتمّ عادة، تغليف القرص الصلب بوعاء من الهواء المضغوط لحمايته من البيئة.

الشكل 4.4 على خلاف القريص المرن، إن القرص الصلب يدور بشكل مستمر وثابت وينتج عن ذلك انخفاض زمن النشد. ونظراً للسرعة الفائقة لدوران القرص الصلب، يقل معدل التأخير الدوراني أيضاً.

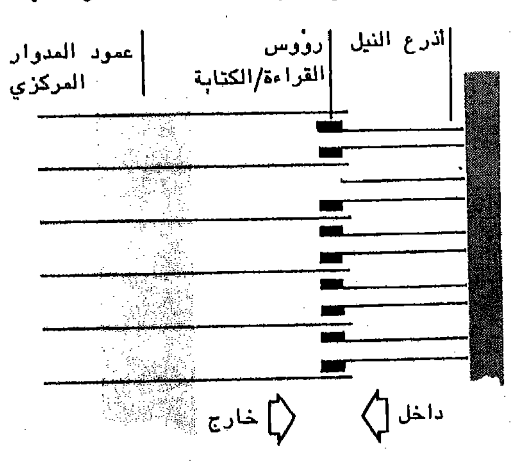


الشكل 4.5 تركب آلية النيل في القرص الصلب على وسادة من الهواء تعلو عن سطح القرص بضعة ملايين من البوصة.

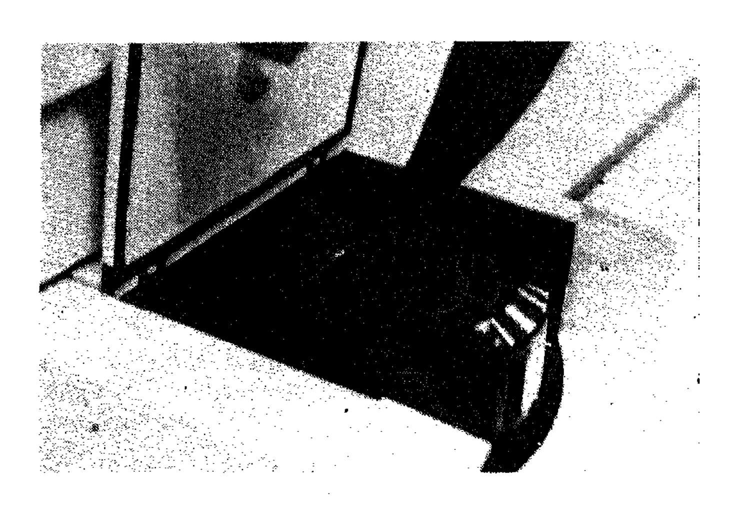


مع توافر اقراص أحادية السطح خصوصاً في الأنظمة ذات الأحجام الصغيرة، تستخدم معظم الكمبيوترات ذات الأحجام الكبيرة مجموعات الأقراص التي تتألف من سطوح تسجيل عديدة مكدسة على عمود مدوار مشترك (شكل 4.6). وبصورة نموذجية، لكل سطح رأسه الخاص بالقراءة والكتابة. وتكون هذه الرؤوس مصفوفة على آلية نيل أحادية تشبه المشط، وتتحرك كلها معاً. على سبيل المثال، لنتصور ان آلية النيل مركزة على السكة 30. سيقوم أعلى رأس القراءة والكتابة بديل السكة 30 على السطح 0. وبتحركه الى أسفل سطحاً سطحاً، سيتركز الرأس الثاني فوق السكة 30 على السطح 1، والراس الثالث فوق السكة 30 على السطح 2 وهلم جراً. وكل موقع لآلية النيل يتطابق مع سكة واحدة على كل سطح. وتسمى مجموعة السكك هذه **بالاسطوانة**.

الشكل 4.6 على مجموعة اقراص، يكون لكل سطح راس قراءة/كتابة خاص به. وتكون هذه الرؤوس مصفوفة على آلية نيل أحادية تشبه المشط. هكذا تتحرك كلها معاً.



الشكل 4.7 تكون بعض مجموعات الأقراص ثابتة بينما يمكن تغيير بعضها الآخر. يقوم هنا، مشغل الكمبيوتر بإزالة مجموعة أقراص من المدوار. بالتالي، سيتمّ تركيب مجموعة جديدة على المدوار نفسه، الأمر الذي يوفّر للكمبيوتر المزيد من المعطيات.



تبدأ طريقة النيل للقرص بزمن النشد. فتتحرّك آلية النيل نحو أسطوانة سبق اختيارها، ويتمّ تشغيل رأس قراءة وكتابة سبق اختياره أيضاً. إن النظام يتطلّع الآن الى سكة أحادية. وفي الخطوة التالية، تدور المعطيات المطلوبة على رأس القراءة والكتابة - هنا يحدث التأخير الدوراني. وأخيراً، تتم قراءة المعطيات، ويتمّ نقلها الى الكمبيوتر.

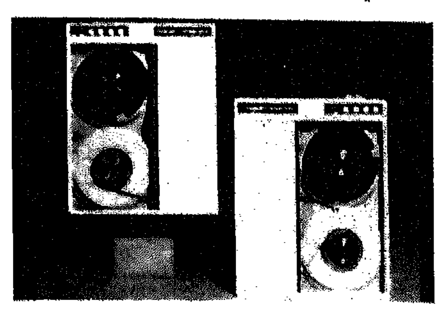
يتم تركيب مجموعة أقراص، كما هو مبيّن في الشكل 4.7، على مدوار قرص يحتوي على محور دوران وعلى آلية نيل. والواقع أنه يمكن تغيير مجموعات الأقراص. فإذا تمّ تفكيك مجموعة أقراص، من الممكن نيل معطيات مجموعة أقراص مختلفة عبر مجموعة رؤوس القراءة والكتابة نفسها. وتوفّر تقنية ونشستر الحديثة مجموعات الأقراص وآليات النيل مختومة معاً في علب مسدودة للهواء مما يعطي كل مجموعة أقراص مجموعة خاصة بها من رؤوس القراءة والكتابة. والجدير بالذكر أن أقراص ونشستر شائعة الاستعمال مع أنظمة الكمبيوتر الأصغر حجماً.

وكما نعلم ان لمجموعات الأقراص سعة كبيرة من المعطيات، فأي فقدان لاحداها بسبب خطأ ما يرتكبه الانسان، أو بسبب حريق أو فيضان أو كوارث مماثلة، يمكن أن يتلف مقداراً كبيراً من المعطيات المهمّة. وفي معظم مراكز الكمبيوتر الضخمة، يتم إجراء خزن احتياطي للمعطيات عن طريق نسخها. وفي حال فقدان مجموعة أقراص، تستخدم النسخة الاحتياطية لاستعادة المعطيات المفقودة.

وسائط أخرى للخزن المساعد

إن **الشريط المغنطيسي** (شكل 4.8) هو من أحد أكثر الوسائط المساندة انتشاراً، ويتسم بسرعة الخزن وبمعدل نقل للمعطيات بالمقارنة مع القرص. أمّا سعة الخزن لديه فهي كبيرة للغاية، والبكرة زهيدة الثمن نسبياً. وللأسف أنه يمكن قراءة المعطيات من، أو كتابتها على، الشريط المغنطيسي، بتسلسل ثابت فقط، الأمر الذي يحصر استخدامه في بضعة مجالات تطبيقيّة.

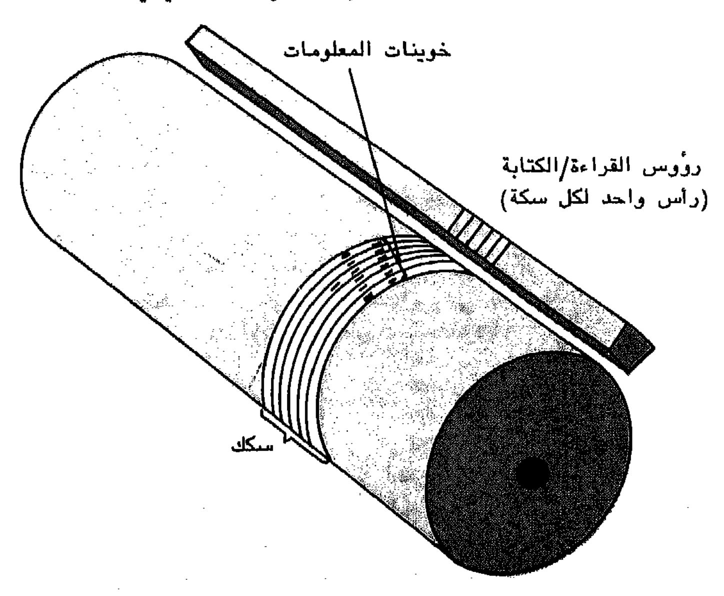
الشكل 4.8 إن الشريط المغنطيسي هو من أحد أكثر الوسائط المساندة انتشاراً.



ان أولى أجهزة الخزن المساعد كانت **الاسطوانة المغنطيسية** (الطبلة الممغنطة (شكل 4.9). وكما يشير اليه الاسم، الطبلة هي أسطوانة تغلّف سطحها الخارجي المادة المغنطيسية نفسها التي تغلف القرص والشريط. وتخزّن المعطيات على سكك متوازية تحيط بالسطح. كما أن لكل سكة رأس قراءة/كتابة خاصاً بها. وبما أن المطلوب ليس تحرّك الرأس، فلا وجود لزمن النشد. وبالتالي تكون الاسطوانة المغنطيسية ذات سرعة فائقة. مع ذلك، وبالمقارنة مع القرص، فللأسطوانة المغنطيسية سعة خزن محدودة، وثمنها مرتفع جداً.

أما القرص البصري فهو أحدث وسائط الخزن المساعد. وتتم القراءة من، والكتابة على هذه الأقراص بواسطة أشعة الليزر، ولا وجود لتلامس مادي بين سطح التسجيل وآلية القراءة والكتابة. وبما أن للقرص البصري ميزات السرعة والدقة والتحكّم بالدمج وسهولة الاستخدام، فتعلّق عليه آمال كبرى في المستقبل.

الشكل 4.9 إن أول جهاز للخزن المساعد هو الاسطوانة المغنطيسية.

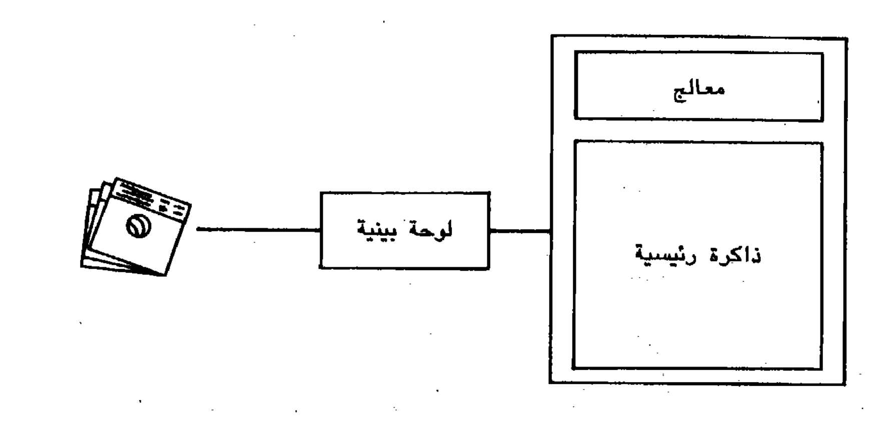


نيل الخزن المساعد

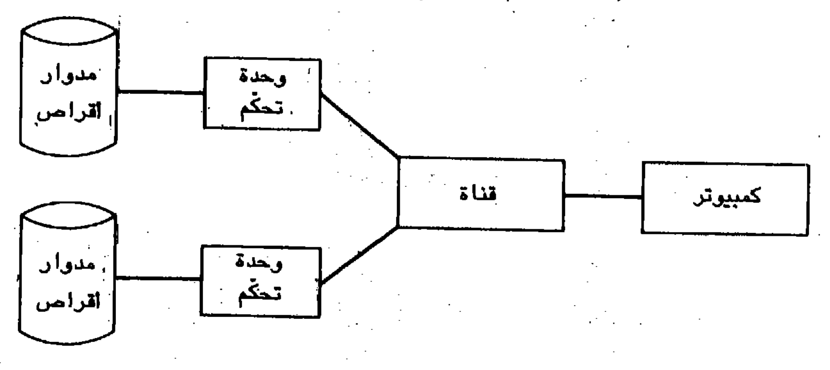
في الكمبيوترات ذات الأحجام الصغيرة، توصل أجهزة الخزن المساعد بالأنظمة عبر لوحات بينية (شكل 4.10). وتتحكّم اللوحة البينية مادياً بمدوار القرص مستقبلة أوامر النشد والقراءة والكتابة من المعالج، وتقوم بتحديد موقع آلية النيل، وتتحكم بتدفّق المعطيات بين سطح القرص والذاكرة الرئيسية. أمّا في الكمبيوترات ذات الأحجام الكبيرة، فتستخدم القنوات ووحدات التحكّم بالدخل/الخرج (شكل 4.11). وتجري القناة اتصالها مع الكمبيوتر بينما تجري وحدة التحكم اتصالها مع الجهاز الخارجي.

ونظراً الى قدرة القرص الأحادي على التخزين، يستطيع أن يخزّن مئات البرامج أو المعطيات، لعشرات من التطبيقات المختلفة. فإذا كنت من مستخدمي الكمبيوتر ورغبت باستخدام برنامج معين، وبنيل مجموعة معيّنة من المعطيات، فكيف يتمكّن الكمبيوتر من ايجاد البرنامج الصحيح أو المعطيات الصحيحة لذلك؟

الشكل 4.10 في الكمبيوترات ذات الأحجام الصغيرة، تكون اجهزة الخزن المساعد موصولة بالأنظمة عبر لوحات بينية.



الشكل 4.11 في الكمبيوترات ذات الأحجام الكبيرة، تكون أجهزة الخزن المساعد موصولة بالكمبيوترات عبر قنوات ووحدات تحكم بالدخل/الخرج.

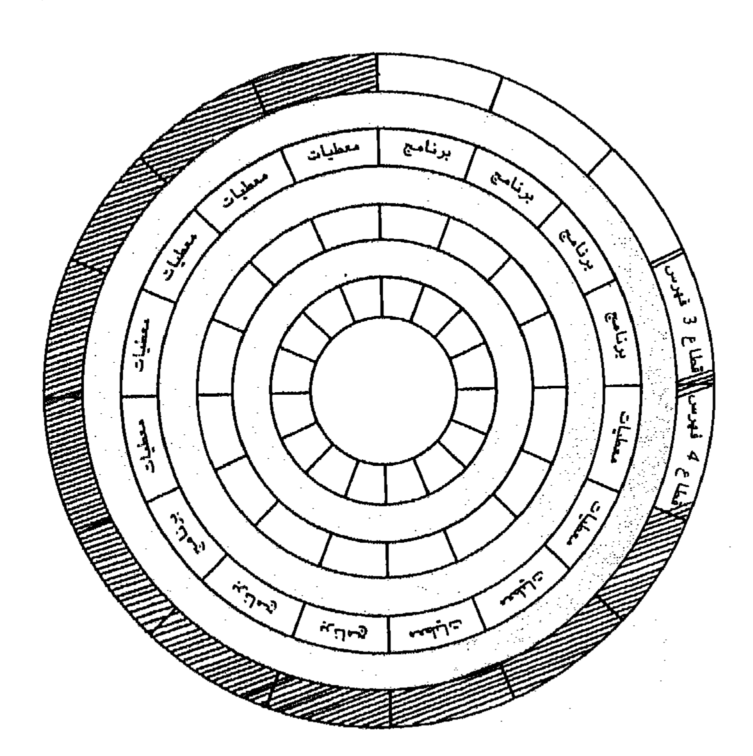


إبدأ بمراجعة كيفية خزن المعطيات على قرص. ينقسم السطح الى سكك تكون بدورها مقسّمة الى قطاعات. وتخزّن المعطيات في القطاعات كأنماط خوينات. وترقم السكك تسلسلياً بحيث يكون رقم السكة الخارجية 0. وفي أثناء تحرّكنا نحو وسط القرص تليها السكة رقم 1 والسكة رقم 2 وهلمّ جرا. كما ترقم القطاعات أيضاً تسلسلياً فيكون رقم القطاع الأول على سكة ما، 0، ورقم القطاع الثاني 1، ورقم القطاع الثالث 2 الخ. ويشكّل القطاع رقم 8 على السكة رقم 5 قطاعاً معيناً، والقطاع رقم 9 على السكة رقم 5 قطاعاً مختلفاً، والقطاع رقم 8 على السكة رقم 6 قطاعاً مختلفاً آخر. وكل قطاع على القرص له عنوان قطاع/سكة منفرد.

وعندما يتمّ خزن برنامج ما على قرص، يسجّل عادة في مجموعة قطاعات متتالية. فاذا بدأ البرنامج على السكة رقم 3، قطاع رقم 0، يمكنك أن تفترض أنه متواصل على السكة رقم 3، قطاع رقم 1 وهلمّ جرا. وإذا استطاع الكمبيوتر أن يجد بداية برنامج ما، يمكنه أن يجد البرنامج بكامله.

كيف يستطيع النظام أن يحدّد موقع بداية برنامج معيّن على القرص؟ عادة، توضَع عدة قطاعات على السكة الأولى جانباً لتحتفظ **بفهرس او دليل** (شكل 4.12). فعندما يتمّ البدء بكتابة برنامج على قرص، يُعطى هذا البرنامج اسماً يتم تسجيله في الفهرس، ويتم تسجيل السكة وعنوان القطاع حيث يبدأ. وعند استرجاعه في وقت لاحق، يقوم مستخدم الكمبيوتر بإدخال اسمه فيقرأ الجهاز الفهرس مفتّشاً عن الاسم، ويجد العنوان حيث يبدأ البرنامج ثم يقوم بقراءته.

الشكل 4.12 يحفظ الفهرس ليتابع تطوّرات البرامج، وملفات المعطيات المخزونة على قرص.



أمّا المعطيات فيتم نيلها بالطريقة نفسها تقريباً. ثم تجمع المعطيات لتطبيق ما لتشكّل ملفاً. وسنقوم بشرح المزيد عن ملفات المعطيات في الفصل الثامن. يتمّ تعيين اسم لكل ملف، ويسجّل كل من اسم الملف وعنوان قطاعه الأول في فهرس القرص. وبما أن المعطيات التي تشكّل ملفاً تكون عادة مخزونة في قطاعات متتالية، فإن معرفة عنوان القطاع الأول تسمح للنظام بإيجاد عناوين القطاعات الأخرى.

إن الخزن المساعد هو امتداد للذاكرة الرئيسية وليس بديلاً عنها. ولا يستطيع الكمبيوتر أن يقوم بتنفيذ برنامج ثم خزنه على قرص إلاّ إذا تمّ نسخه أولاً في الذاكرة الرئيسية. كما لا يستطيع أن يعالج المعطيات التي تمّ خزنها في وسيط ثانوي حتى يتمّ نسخها في الذاكرة الرئيسية. وتحتفظ الذاكرة الرئيسية بالبرامج والمعطيات الحالية بينما يستخدم الخزن المساعد لخزن تطول مدّته.

إن أجهزة الدخل والخرج التي ورد وصفها في الفصل 3 تقدّم للانسان ميزة نيل نظام الكمبيوتر، بينما لا يوفر الخزن المساعد ذلك. وتكون المعطيات مخزونة بالشكل الذي يناسب الكمبيوتر، وتتمّ قراءتها أو كتابتها عن طريق الآلة فقط. والطريقة الوحيدة التي تسمح للناس بنيل المعطيات المخزونة على قرص هي بإدخال تعليمة الى الكمبيوتر لقراءة هذه المعطيات من الذاكرة الرئيسية ثم خرجها الى شاشة عرض، أو الى طابعة.

الخلاصة

بدأ هذا الفصل ببحث موجز عن الحاجة الى الخزن المساعد. ونظراً الى كلفة الذاكرة الرئيسية وسعتها المحدودة وعدم استقرارها، لا يمكن استخدامها في خزن المعطيات لمدة طويلة، أو في خزن كميات كبيرة من المعطيات. لذلك، يكمن الحل باستخدام الخزن المساعد.

إن الكاسيت المغنطيسي هو الأرخص ثمناً في وسائط الخزن المساعد، والقريص المرن هو أكثرها استعمالاً في الميكروكمبيوترات. وتخزّن المعطيات على سطح القرص في سلسلة من الدوائر المتحدة المركز التي تُسمّى سككاً. وتنقسم هذه السكك الى قطاعات. فعند الدخل، يتم نسخ محتويات قطاع واحد من القرص الى الذاكرة الرئيسية، وعند الخرج، ينتقل قطاع واحد من الذاكرة الى سطح القرص. ولكي تتمّ عملية النيل من القرص، يجب أولاً تشغيل مدوار الأقراص حتى يبلغ سرعة التشغيل التي تمكن آلية النيل من التحرك فوق السكة التي تحتوي على المعطيات المطلوبة (زمن النشد). ويضيع الزمن الاضافي بانتظار القطاع المطلوب نيله ليدور حتى يبلغ السرعة التي تمكّنه من ملامسة رأس القراءة والكتابة (التأخير الدوراني).

وخلافاً لدوران القريص المرن، يدور القرص الصلب على نفسه بسرعة ثابتة، هكذا، يتمّ إزالة مكوّن رئيسي من زمن النشد. ويكون معدّل التأخير الدوراني لدى القرص الصلب أقل منه لدى القريص المرن، ذلك لأن سرعة دوران الأول أكبر من سرعة دوران الثاني. كما يتميّز القرص الصلب بسعة خزن أكبر من سعة الخزن لدى القريص المرن. وفي القريص المرن، يركب رأس القراءة والكتابة عادة، بطريقة مباشرة على سطح القرص بينما بالنسبة الى القرص الصلب تركّب آلية النيل على وسادة من الهواء تبعد بضعة ملايين من البوصة عن السطح. وغالباً، تتكدّس عدة سطوح على عامود مدوار أحادي لتشكّل مجموعة أقراص. وتحتوي مجموعة الأقراص عادة، على رأس قراءة وكتابة واحد لكل سطح، وتتجمّع رؤوس القراءة والكتابة على آلية نيل أحادية. وبالتالي يحدّد موضع آلية النيل سكة واحدة على كل سطح من سطوح القرص، وتُسمّى مجموعة السكك هذه بالأسطوانة.

ونظراً الى اعتبار المعطيات قيّمة للغاية، يتمّ عادة الاحتفاظ بمجموعات الأقراص على وسائط خزن احتياطية. والشريط المغنطيسي وسيط خزن شائع. أمّا الأسطوانة

المغنطيسية (الطبلة الممغنطة) فهي من أولى وسائط الخزن الاحتياطي المستخدمة في الكمبيوتر. ويُعدّ القرص البصري أيضاً من وسائط الخزن المشجّعة للاستخدام في المستقبل.

في أنظمة الكمبيوتر ذات الحجم الصغير، يتمّ وصل أجهزة الخزن المساعد الى الكمبيوتر عبر لوحات بينية. أمّا في الأنظمة ذات الحجم الأكبر، فتضبط وحدة التحكم بالدخل/الخرج مدوار الأقراص ضبطاً مادياً بينما تقوم قناة بإجراء الاتصال مع الكمبيوتر.

يستطيع قرص أحادي أن يحتوي على برامج وملفات معطيات عديدة. ولتمييزها، يتمّ الاحتفاظ بفهرس ليقوم بتحديد مواقع البرامج ومواقع ملفات المعطيات عن طريق الاشارة الى السكك وعناوين القطاعات التي يبدأ عندها كل منها. وعند معرفة عنوان القطاع الأول يسهل تحديد مواقع القطاعات الأخرى.

يُعَدّ الخزن المساعد امتداداً للذاكرة الرئيسية. فلا يستطيع جهاز الكمبيوتر أن ينفذ برنامجاً ما حتى يكون قد تمّ تحميله في الذاكرة الرئيسية، ولا يستطيع أن يعالج المعطيات حتى يكون قد تم نسخها في الذاكرة الرئيسية. وليس باستطاعة الانسان أن يقرأ وسائط الخزن المساعد لأنها وسيط قابل للقراءة من قبل الآلة فقط.

مصطلحات أساسية

- ☐ آلية النيل
- ☐ احتياطي
- ☐ كاسيت
- ☐ أسطوانة
- ☐ دليل
- ☐ قرص
- ☐ مجموعة اقراص
- ☐ قريص مرن
- ☐ قرص مرن
- ☐ قرص صلب
- ☐ فهرس
- ☐ أسطوانة مغنطيسية (طبلة ممغنطة)
- ☐ شريط مغنطيسي
- ☐ تاخير دوراني
- ☐ خزن مساعد
- ☐ قطاع
- ☐ زمن النشد
- ☐ سكة
- ☐ قرص بصري

اختبار ذاتي

1. أي من الآتي لا يُعَد مشكلة في الذاكرة الرئيسية؟

   أ. عدم الاستقرار
   ب. السعة المحدودة
   ج. الكلفة
   د. السرعة

2. أكثر وسائط الخزن المساعد المستخدمة في الميكروكمبيوتر هي:

   أ. القريص المرن
   ب. القرص الصلب
   ج. الشريط المغنطيسي
   د. القرص البصري

3. تسجّل المعطيات على القرص في سلسلة من الدوائر المتحدة المركز التي تُسمّى بـ__________.

   أ. قطاعات
   ب. أسطوانات
   ج. قوالب
   د. سكك

4. تنتقل المعطيات بين الذاكرة الرئيسية وسطح القرص بمعدل ____________ في كل مرة.

أ. سكة
ب. أسطوانة
ج. قطاع
د. كلمة

5. إن التأخير الزمني الذي تزداد خلاله سرعة مدوار القرص لتبلغ معدل سرعة التشغيل الذي يسمح لآلية النيل بأن تحدّد موقعها هو ____________.

أ. زمن النيل
ب. التأخير الدوراني
ج. زمن النشد
د. فترة غير إنتاجية

6. إن الفترة الزمنية التي يستغرقها دوران القطاع المطلوب ليبلغ آلية النيل تُسمّى ____________.

أ. التأخير الدوراني
ب. زمن النيل
ج. زمن التنفيذ
د. زمن النشد

7. يدور ____________ بسرعة مستمرة وثابتة.

أ. مدوار القريص المرن
ب. مدوار الكاسيت
ج. القرص الصلب
د. مدوار القرص المرن

8. على القرص الصلب، يكون التأخير الدوراني أقل منه على القريص المرن لأن سرعة الدوران ____________.

أ. تحذف
ب. لا تكون موجودة
ج. تزداد
د. تنقص

9. تشكّل عدة أقراص مكدّسة على محور مدوار أحادي ____________

أ. أسطوانة
ب. أسطوانة (طبلة)
ج. مجموعة أقراص
د. قرصا بصريا

10. علي مجموعة أقراص، يقوم موضع واحد من آلية النيل بنيل مجموعة سكك تُسمى ____________.

أ. كلمة
ب. قطاعا
ج. قالبا
د. أسطوانة

11. إن وسيط الخزن الاحتياطي الذي يكثر استخدامه هو ____________.

أ. القريص المرن
ب. الشريط المغنطيسي
ج. الأسطوانة المغنطيسية (الطبلة الممغنطة)
د. القرص البصري

12. في الكمبيوتر ذي الحجم الصغير، يتمّ وصل أجهزة الخزن المساعد الى النظام عبر ____________.

أ. وحدات التحكّم ج. لوحات بينية
ب. قنوات د. كيانات منطقية

13. في أنظمة الكمبيوتر ذات الأحجام الكبيرة، أي ممّا يلي يجري الاتصال مع مدوار الأقراص؟

أ. القناة ج. وحدة التحكّم بالدخل/الخرج
ب. اللوحة البينية د. المعالج

14. إن تركيبة رقم قطاع مع رقم سكة تشكّل ______________ منفرداً.

أ. فهرساً ج. عنوان قرص
ب. إسم برنامج د. إسم ملف

15. يوجد عنوان لبداية كل برنامج تمّ خزنه على قرص في ______________ القرص.

أ. برنامج ج. فهرس
ب. أداة تحكّم د. بينية

الاجابات

1.د 2.أ 3.د 4.ج 5.ج 6.أ 7.ج 8.ج 9.ج 10.د 11.ب 12.ج 13.ج 14.ج 15.ج.

ربط المفاهيم

1. لماذا يُعتبر الخزن المساعد ضروريا؟
2. تصوّر مجموعة معطيات تمّ خزنها في الذاكرة الرئيسية. إشرح عملية نقل هذه المعطيات الى القرص. تكلم بالتحديد واشرح كل خطوة في العملية.
3. ما هو الفرق بين قريص مرن وقرص صلب؟ وما هي الفوائد من استخدام القرص الصلب؟
4. ما هي مجموعة الأقراص
5. ميّز بين أسطوانة، وسكة، وقطاع.
6. ما هو الخزن الاحتياطي؟ ولماذ يُعد ضروريا؟
7. كيف تكون أجهزة الخزن المساعد موصولة مادياً بالكمبيوتر؟ وكيف تكون أجهزة الدخل والخرج موصولة بالكمبيوتر؟ قارن بين الاجابتين.
8. إشرح كيفية إيجاد الكمبيوتر للمعطيات «الصحيحة»، أو للبرامج «الصحيحة» التي تمّ خزنها على القرص.
9. الخزن المساعد هو امتداد للذاكرة الرئيسية. إشرح ذلك.

10. في الفصل 1 (التمرين 5) قمت برسم نظام كمبيوتر بسيط، وفي الفصل 2 (التمرين 9) أضفت المزيد من التفاصيل على هذا الرسم. أمّا في الفصلين 3 و4 فتعلّمت كيفية توصيل أجهزة الدخل والخرج وأجهزة الخزن المساعد الى نظام الكمبيوتر. فمرة أخرى، قم بتعديل الرسم موضحاً كيفية توصيل الأجهزة المحيطية مادياً. قد تودّ أن تعد رسمين، واحداً لميكروكمبيوتر وآخر لكمبيوتر رئيسي.

5.

وصل المكوّنات

مفاهيم أساسية

الميكروكمبيوترات والكمبيوترات الرئيسية
- □ خطوط الناقلات والكبلات
- □ حجم الكلمة

تصميم الميكروكمبيوتر

تصميم الكمبيوتر الرئيسي

الميكروكمبيوترات والكمبيوترات الرئيسية

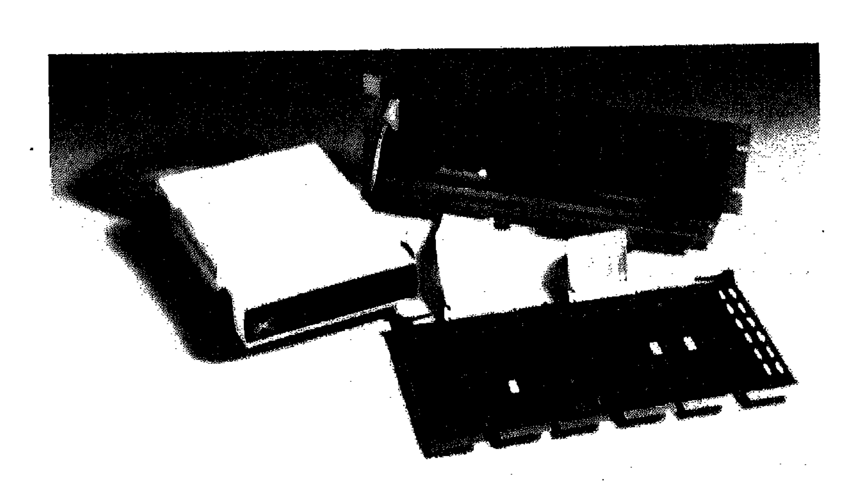
حتى هذه النقطة درسنا مكوّنات الكمبيوتر الأولية الواحد تلو الآخر. انت تعرف أن الكمبيوتر يحتوي على معالج، وذاكرة رئيسية، وعدد من المراصف، وانه يمكن أن تلحق به أجهزة دخل وخرج وخزن ثانوي عديدة. ولكن ربما بدا لك أن تنظر الى هذه المكوّنات كصناديق مستقلة. وقد حان الوقت للوصل بين القطع. وفي هذا الفصل سنبحث عدداً من الطرق التي يمكن بها تجميع مكونات الكمبيوتر.

خطوط الناقلات والكبلات

الكمبيوتر نظام تتدفّق المعطيات بين مكوّناته استجابة لتعليمات ينفّذها المعالج. وفي بعض الأحيان تُنقل المعطيات **بالتوازي**؛ وتتواصل المكوّنات الأخرى **تسلسلياً**. تصور جسراً ذا قناة واحدة تعبره السيارات في صف واحد. والآن، تصوّر جسراً به أربع قنوات تنطلق السيارات على كل منها. فمن الواضح ان الجسر ذا القنوات الأربع يستطيع أن يمرّر عدداً أكبر من السيارات في فترة زمنية معيّنة. وهذه القاعدة العامة نفسها تنطبق على خطوط النقل المتوازية والتسلسلية. ويتألف الخط المتوازي من عدد من الأسلاك ومن ثم فإنه يستطيع أن ينقل العديد من الخوينات في كل مرة. ويتألف الخط التسلسلي من سلك أحادي، ولذلك يمكن نقل خوينة واحدة فقط في كل مرة. ان الخطوط المتوازية أسرع.

أمّا خطوط الناقلات، مثل مجموعة الأسلاك المتوازية التي تشبه الشريط والمبيّنة في الشكل 5.1، فإنها تربط بين مكوّنات الكمبيوتر الداخلية وتُستعمل لالحاق أجهزة خزن ثانوية بالنظام. انها تنقل المعطيات بالتوازي. ويجري وصل أجهزة الدخل والخرج بالنظام بواسطة الكبلات. وتتألف الكابلات المتوازية من اسلاك عديدة؛ امّا الكبلات التسلسلية فتتألف من سلك أحادي وتُستعمل لتلحق بالكمبيوتر اجهزة محيطية أبطأ.

الشكل 5.1 مكوّنات الكمبيوتر الداخلية موصولة بخطوط ناقلات شبيهة بالأشرطة. وهناك اجهزة خارجية ملحقة بالنظام بواسطة كبلات.



حجم الكلمة

ان الاتصال بين المكوّنات يمكن أن يبسط الى حدّ كبير اذا كانت متشابهة الكترونياً. وهكذا، وفي معظم الأنظمة، يصمّم المعالج، والذاكرة الرئيسية، والناقلات، والبينيات، والقنوات جميعها وفق حجم **كلمة** مشترك. على سبيل المثال، في كمبيوتر ذي 32 خوينة، يعد المعالج اعداداً ذات 32 خوينة، وذاكرة رئيسية وتخزّن المراصف كلمات ذات 32 خوينة وتتحرّك المعطيات والتعليمات بين المكوّنات عبر خطوط ناقل ذي 32 خوينة. وتقريباً فإن حجم أية كلمة يفي بالغرض رغم ان الأحجام 4، 8، 16، 32 خوينة هي الأكثر شيوعاً.

ويؤثر حجم كلمة الكمبيوتر في سرعة معالجته وسعة ذاكرته ودقّته وتكلفته. دعنا ننظر الى السرعة أولاً. فالناقل ذو الـ32 خوينة يحتوي على 32 سلكاً ومن ثم يستطيع أن ينقل 32 خوينة في المرة الواحدة. يملك الناقل ذو الـ16 خوينة 16 سلكاً متوازياً فقط ومن ثم فانه ينقل 16 خوينة فقط. ولأن الناقل الأعرض يحرك من المعطيات ضعف ما يحرّكه الناقل الآخر في المدة نفسها، فمن الواضح ان الآلة ذات الـ32 خوينة أسرع. وعموماً فكلما كان حجم الكلمة أكبر كان الكمبيوتر أسرع.

إن سعة الذاكرة، هي أيضاً، من وظائف حجم الكلمة. وعلى المعالج، لكي ينال ذاكرة رئيسية، أن يرسل عبر الناقل عنوان التعليمة المرغوب فيها أو عنصر المعطيات. وفي آلة ذات 32 خوينة يمكن ارسال عنوان ذي 32 خوينة. وأكبر عدد ذي 32 خوينة يعادل بالتقريب 4 بلايين بالتعبير العشري، وعلى هذا فان المعالج يستطيع أن ينال حتى 4 بلايين موقع مختلف في الذاكرة. ومن الناحية الأخرى فان الكمبيوتر ذا الـ16 خوينة يرسل عنواناً ذا 16 خوينة، محدّداً نيله بـ64000 موقع ذاكرة تقريباً. وعموماً، كلما كان حجم كلمة الكمبيوتر أكبر، فان بامكانه أن ينال أكثر من الذاكرة الرئيسية.

وهناك ميكروكمبيوترات ذات 16 خوينة تنال بشكل ملموس أكثر من 64 كيلو خانة من الذاكرة. كيف يمكن ذلك؟ تستطيع آلة ذات 16 خوينة أن تنال أكثر من 64 كيلو إذا قسمت العناوين الى جزئين أو أكثر وأرسلت في أثناء دورات الآلة المتتالية. على ان كل دورة تستغرق وقتاً، ولذلك تكسب سعة الذاكرة على حساب سرعة المعالجة.

وبعد، لننظر الى حجم الأعداد التي تستطيع كل آلة أن تعالجها. يعمل كمبيوتر ذو 16 خوينة باعداد ذات 16 خوينة؛ وتعمل آلة ذات 32 خوينة باعداد ذات 32 خوينة. والأرقام الأكثر تعني اجابات أدق. ويستطيع العديد من الكمبيوترات ذات الـ١٦ خوينة أن تحول وتعمل باعداد ذات 32 خوينة، ولكنها تحتاج الى دورتين لتجلب العدد من الذاكرة ودورتين أخريين لتعالجه. وعلى آلة أصغر يمكن تحقيق الدقّة على حساب سرعة المعالجة كما هو الحال بالنسبة الى سعة الذاكرة.

وتستعمل المعالجات الميكروية ذات الـ4 خوينات والمنخفضة التكلفة، في مجموعة منوّعة من المنتجات الاستهلاكية بما في ذلك السيارات والأدوات المنزلية وحتى لعب الأطفال. وقد صمّم الكثير من **الميكروكمبيوترات** رخيصة الثمن على أساس كلمات ذات 8 خوينات؛ ويحدّ حجم الكلمة الصغير هذا من سعة ذاكرتها وسرعة معالجتها ودقّتها (شكل 5.2). وتستعمل الميكروات الأغلى ثمناً ومعظم **المينيكمبيوترات** كلمة ذات 16 خوينة معطية اياها سرعة أكبر، ومزيداً من الذاكرة الرئيسية وحتى مزيداً من الدقّة. وتقوم **الكمبيوترات الرئيسية** نموذجياً على كلمة ذات 32 خوينة، وتبعاً لذلك فهي أكبر وأسرع وأكثر دقّة. وتملك بعض «السوبركمبيوترات» العلمية كلمة ذات 60 أو 64 خوينة.

يؤثر حجم الكلمة أيضاً على تكلفة النظام. ومعظم الآلات ذات الـ8 خوينات يقلّ سعرها كثيراً عن 1000 دولار في حين ان الميكروات الأقوى ذات الـ16 خوينة تُباع بـ2000 دولار أو أكثر (شكل 5.2). يكلّف المينيكمبيوتر النموذجي نحو 10,000 دولار،

في حين ان الكمبيوتر الرئيسي قد يصل سعره الى 100,000 دولار وقد يزيد سعر «الماكسي» بسهولة عن مليون دولار. وهناك قدر كبير من التداخل، اذ تنافس «المينيكمبيوترات» ذات الـ16 خوينة رخيصة الثمن «السوبركمبيوترات» ذات الـ32 خوينة و«السوبرميني» التي تؤدّي تقليدياً مهام الكمبيوترات الرئيسية. ان مصطلحات ميكرو وميني ورئيسي هي فقط خطوط هادية وليست ثوابت.

ويلاحَظ انه عند مناقشة عوامل، مثل حجم الكلمة، وعرض الناقل، والسرعة وسعة الذاكرة، والدقّة والتكلفة، يسهل اغفال ما هو واضح. الميكروكمبيوترات (شكل 5.3) صغيرة وتملك عدداً محدوداً من الأجهزة المحيطية. ومن الناحية الأخرى، فان الكمبيوترات الرئيسية (شكل 5.4) أكبر بكثير ونموذجياً فإنها تتحمل مجموعة كبيرة من أجهزة الدخل والخرج والخزن الثانوي.

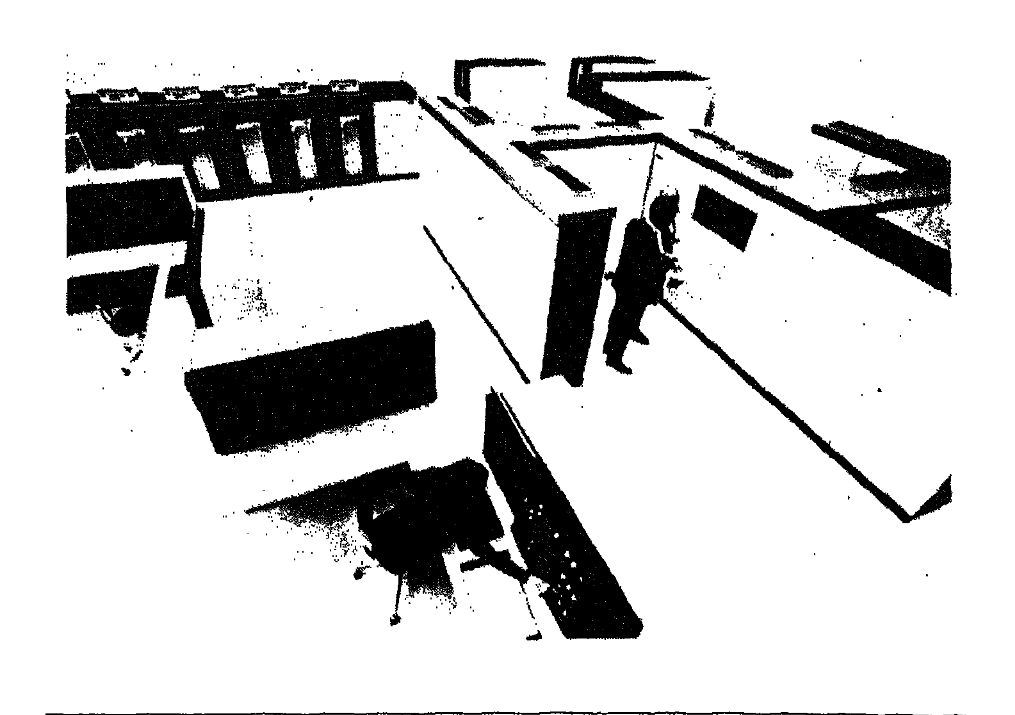
الشكل 5.2 مقارنة بين انواع الأنظمة المختلفة.

نوع الكمبيوتر	حجم الكلمة	زمن الدورة	سعة الذاكرة	التكلفة بالدولار	امثلة
ميكروكمبيوتر منزلي	8 خوينات	ميكروثوان	64 ك. خانة	حوالي 1000 دولار	Apple IIc, Apple IIe
ميكروكمبيوتر مهني	16 خوينة	ميكروثانية	256 ك. خانة	اكثر من 2000 دولار	IBM PC
مينيكمبيوتر	16 خوينة	250 نانوثانية	512 ك. خانة	+ 10 000 دولار	DEC PDP-11
كمبيوتر رئيسي	32 خوينة	50 نانوثانية	4 ميغاخانة	+ 100 000 دولار	IBM 3083, 4361
سوبر كمبيوتر	64 خوينة	10 نانوثانية	4 + ميغاخانة	مليون دولار +	Gray X-MP/2

الشكل 5.3 نظام ميكروكمبيوتر نموذجي.



الشكل 5.4 نظام كمبيوتر رئيسي.



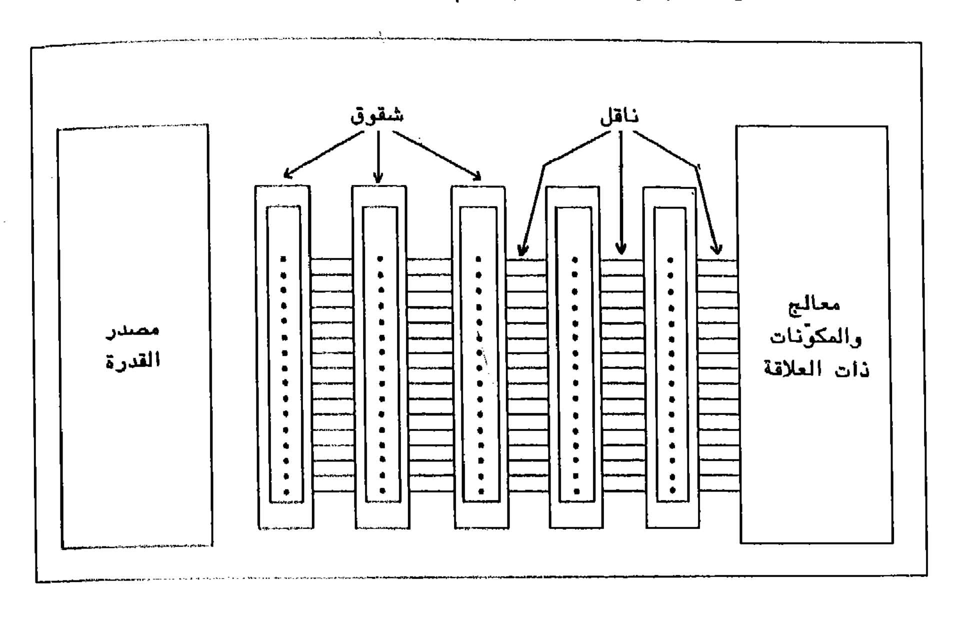
تصميم الميكروكمبيوتر

إذا كشفنا الغطاء عن الميكروكمبيوتر والكمبيوتر الرئيسي، فسنرى انه مع ان كليهما يحتويان على مكوّنات متشابهة، فإن تلك المكوّنات قد جمعت بطرق مختلفة. يستخدم علماء الكمبيوتر المصطلح «**تصميم**» عندما يناقشون العلاقات بين مكوّنات نظام الكمبيوتر. دعنا نتحقّق من الأمر.

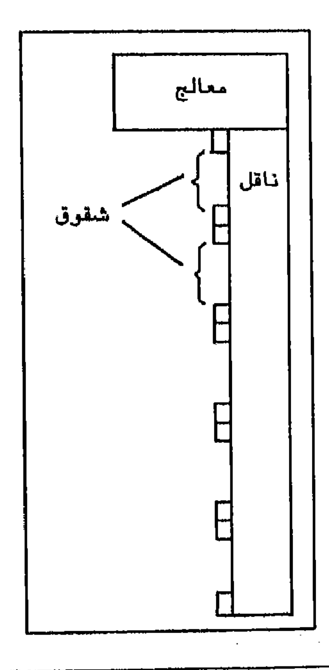
تُباع الميكروكمبيوترات للأفراد حديثي العهد بالكمبيوترات. وعموماً تكون استعمالاتها قليلة وتشمل برامج قصيرة وتعالج قدراً محدوداً من المعطيات. وفي حين لا ينطبق ما ذكرناه إلاّ على عدد قليل من مستخدمي الميكروكمبيوتر، ولكنه تعميم معقول. إن كمبيوتراً مصمّماً بهدف إيجاد سوق رائجة يجب أن يُراعى فيه الكلفة المنخفضة وسهولة الاستعمال والوثوقية فيما يضحى، إذا لزم الأمر، بسرعة المعالجة والدقّة وسعة الذاكرة والأجهزة المحيطية. لذلك تذكر هذه العوامل عندما نفحص في هندسة نظام الميكروكمبيوتر.

تُركّب معظم الأنظمة حول **لوحة أم** (شكل 5.5)، وهي هيكل معدني يحتوي على سلسلة من **الشقوق** الموصولة عبر ناقل الى معالج ذي 8 أو 16 خوينة (شكل 5.6). تُضاف الذاكرة بتوصيل لوحة ذاكرة في أحد الشقوق (شكل 5.7). فبدون الدخل والخرج لا فائدة من الكمبيوتر، ولذلك توصل لوحة مفاتيح/عرض بينية في شق آخر وتوصَل بها وحدة لوحة مفاتيح وعرض (شكل 5.8). ويمكن إلحاق طابعة عبر بينية طابعة، وإذا وصلت بينية قريص في شق مفتوح، يمكن اضافة مدوار قريص. لدينا الآن نظام ميكروكمبيوتر كامل.

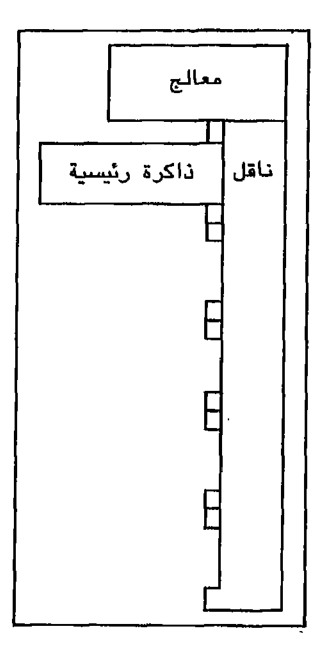
الشكل 5.5 يُركَّب الميكروكمبيوتر حول هيكل معدني يُسمَّى اللوحة الأم. تموذجياً يُركَّب المعالج والمكوّنات ذات الصلة على اللوحة الأم ويوصل ناقل المعالج بسلسلة من الشقوق التي تستعمل لالحاق لوحات أخرى بالنظام.



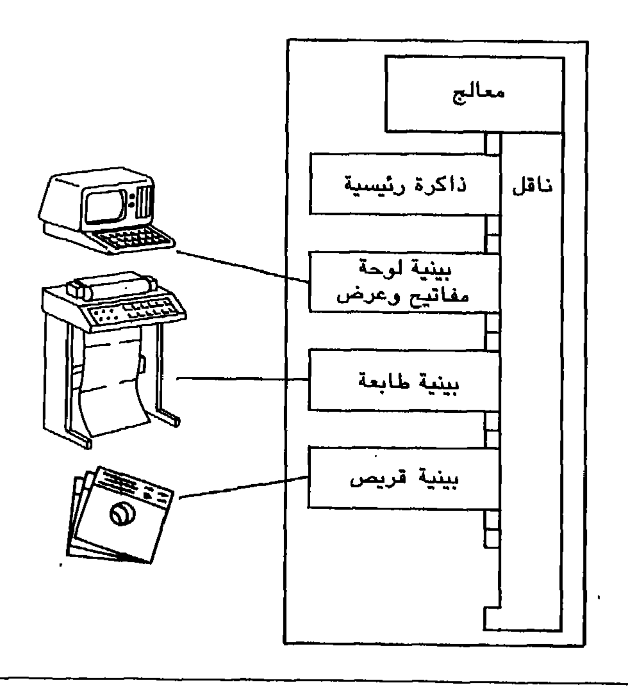
الشكل 5.6 رسم تخطيطي يوضح المعالج واللوحة الأم. يربط ناقل المعالج بعدد من الشقوق التي يمكن أن توصل بها المكونات بالقبس.



الشكل 5.7 توصل لوحة ذاكرة بأحد الشقوق المفتوحة.



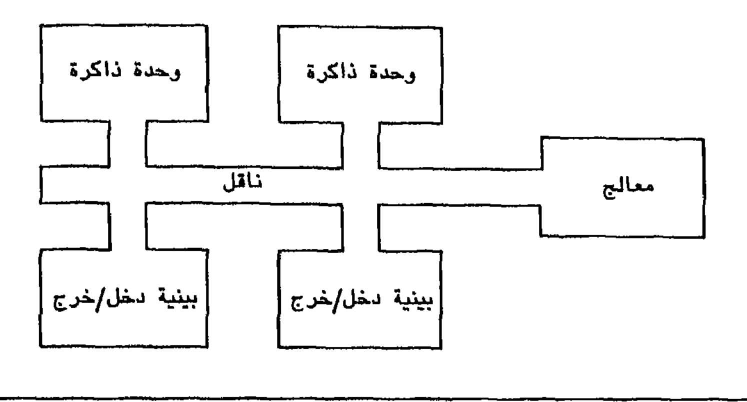
الشكل 5.8 أجهزة دخل وأجهزة خرج وأجهزة خزن ثانوي تُضاف الى النظام بوصل البينيات الملائمة في شق مفتوح ثم تمرير كبل من الجهاز الخارجي الى البينية.



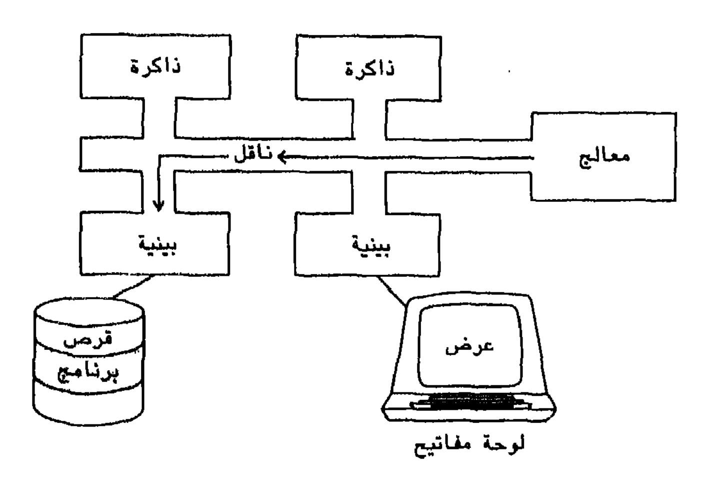
هناك شق مفتوح اضافي في نظامنا الافتراضي. ففي استطاعتنا أن نستعمل هذا الشق لاضافة ذاكرة رئيسية أخرى، أو مدوار قرص آخر، أو طابعة أخرى أو جهاز محيطي آخر، ولكن ليس في استطاعتنا أن نضيف هذه الأجهزة جميعها، لأن عدد الشقوق المتوافرة لدينا يحدّ من توسّع نظام الكمبيوتر. تذكر، من ناحية ثانية طبيعة مستخدمي الكمبيوتر النموذجي. فمعظمهم يطلب سعة محدودة؛ وعلى هذا فإن الحقيقة القائلة بأن التصميم يحدّ من التوسّعية لا تشكّل مشكلة خطيرة. فربما يحتاج فعلاً الراغبون في المزيد من السعة الى مينيكمبيوتر أو كمبيوتر رئيسي.

إن المكونات المرسومة في الشكل 5.8 موصولة بناقل مشترك. ويُسمّى هذا الترتيب تصميم الناقل الأحادي (الشكل 5.9). وتتدفّق كل الاتصالات بين المكونات عبر هذا الناقل، تحت سيطرة المعالج.

الشكل 5.9 الميكروكمبيوتر النموذجي يستعمل تصميم ناقل احادي، مع جميع المكوّنات الداخلية موصولة بخط ناقل احادي.



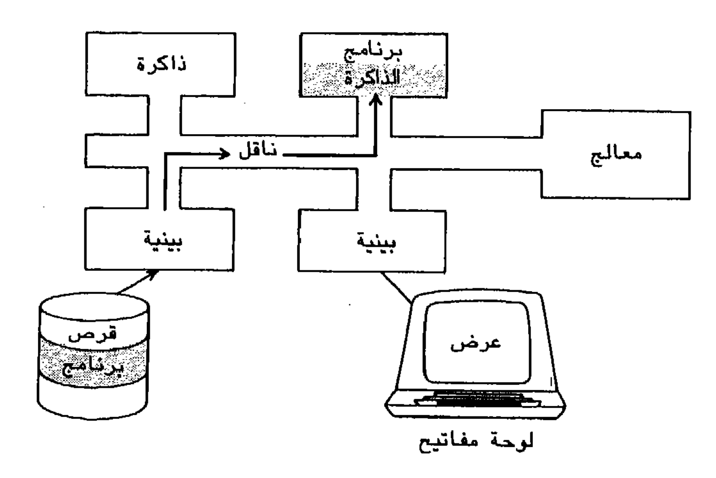
الشكل 5.10 عملية تحميل وتنفيذ برنامج تشتمل على خطوات عديدة.



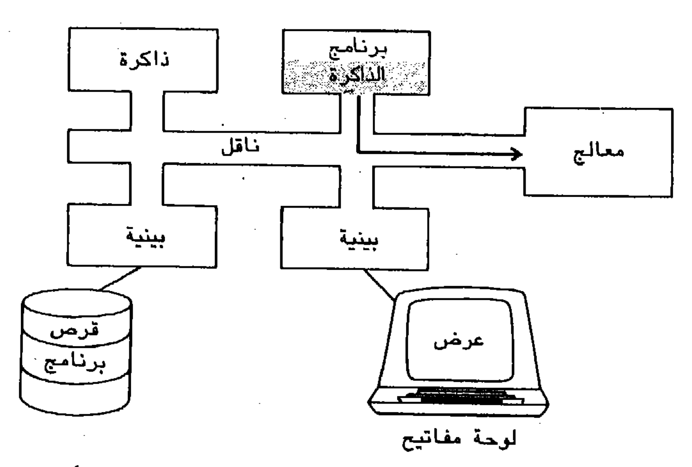
أ. استجابة لطلب المستعمل لتنفيذ برنامج، يرسل المعالج اشارة عبر الناقل الى بينية القرص.

لنبحث مثلاً تحميل وتنفيذ البرنامج. أولاً، تجاوباً مع أمر المستخدم، تُرسل اشارة عبر الناقل الى بينية القرص (شكل 5.10أ). وتتصل بينية القرص استجابة للاشارة، مع مدوار القرص، الذي يقرأ البرنامج ويحوله الى الناقل والى داخل الذاكرة الرئيسية (شكل ب5.10). وبمجرد وصول البرنامج الى الذاكرة، يستطيع المعالج أن ينفذه وذلك بالبحث عن تعليماته الواحدة تلو الأخرى (شكل ج5.10). وبالطبع فإن التعليمات تتحرك من الذاكرة الى المعالج عبر الناقل. وأخيراً، وبينما تنفذ تعليمات البرنامج، يتحرك دخل المعطيات من جهاز محيطي عبر الناقل ثم الى داخل الذاكرة (شكل د5.10)، بينما يتحرك الخرج من الذاكرة عبر الناقل الى جهاز الخرج.

الشكل 5.10

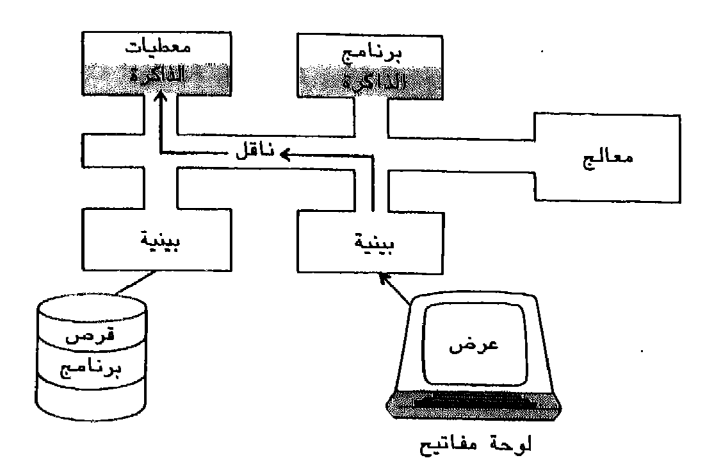


ب. ينقل البرنامج من سطح القرص داخل البينية عبر الناقل الى داخل الذاكرة الرئيسية.



ج. بمجرد وصول البرنامج الى الذاكرة، يستطيع المعالج أن ينفذه، مستحضراً تعليماته الواحد الأخرى عبر الناقل.

الشكل 5.10



. بينما يجري تنفيذ البرنامج، ينتقل دخل المعطيات من جهاز دخل داخل بينية عبر الناقل والى الذاكرة الرئيسية.

لأن الاشارات الالكترونية التي تتحكّم في وحدة لوحة المفاتيح والعرض والطابعة ومدوار القريص مختلفة، فإن لكل جهاز محيطي بينيته الخاصة به. ويتصل جانب واحد بالكمبيوتر، مستخدماً أكواداً داخلية (شكل 5.11). أمّا الجانب الآخر المُعتمد على الجهاز، فيتصل بالجهاز الخارجي بشروطه الخاصة. إن الوظيفة الأساسية للبينية هي الترجمة. وعلى سبيل المثال، يمثل الحرف A بشكل ملموس كمفتاح على لوحة مفاتيح أو كنمط نقطة على طابعة على حدّ سواء. وعندما يطبع المستعمل الحرف A، تدخل نبضة الكترونية بينية لوحة المفاتيح والعرض حيث تترجم الى الكود الثنائي الذي يمثّل A داخل الكمبيوتر. وفيما بعد، يرسل هذا الكود نفسه في الخرج الى بينية طابعة، حيث يترجم الى الاشارات المطلوبة لتشكّل نمط النقطة الصحيح. لاحظ ان الكمبيوتر يستعمل دائماً نفس الكود الثنائي، مهما كان الجهاز المحيطي المتضمّن. وبالنسبة للمعالج فإن جميع الأجهزة المحيطية متشابهة.

الشكل 5.11 الوظيفة الأساسية للبينية هي الترجمة بين اشكال المعطيات الخارجية والداخلية.

شكل الكمبيوتر الداخلي
ترجم
الشكل الخارجي

تصميم الكمبيوتر الرئيسي

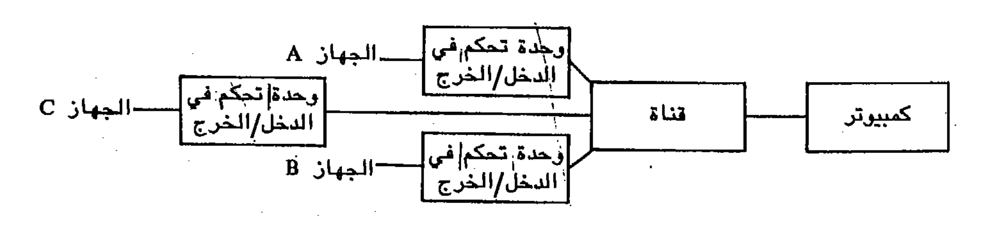
لأن الميكروكمبيوتر مصمّم لمستعمل واحد، فإن تصميم الناقل الأحادي معقول. أمّا الكمبيوتر الرئيسي، على كل حال، فإنه مصمّم لمزيج مختلف من التطبيقات، وغالباً ما يكون النظام مستودعاً لمعطيات المنظمة التشغيلية والتخطيطية الرئيسية. والفكرة هي اتاحة الفرصة لكل المستعملين لنيل قاعدة المعطيات المشتركة الرئيسية هذه. وعندما يكون عدد المستعملين كبيراً وكميات المعطيات هائلة، فإن سعتَي الخزن الأولية والثانوية حاسمتان. وستكون هنالك حاجة الى العديد من أجهزة الدخل والخرج، وتكون البرامج كبيرة ومعقّدة، ولذلك فإن سرعة المعالجة تكون ضرورية أيضاً.

إن نظاماً له معالج ذو 32 خوينة، ومليون خانة أو أكثر في الذاكرة الرئيسية، ومقدار لا حصر له من أجهزة الخزن الثانوية، والعديد من أجهزة دخل/خرج، تكون كلفته عالية، ولذلك تكون فاعليته من اهتمامات الادارة الأساسية. انه لأمر مناف للعقل أن يُجبر كمبيوتر غالي الثمن على الانتظار بينما يقوم انسان بطبع سطر دخل. والحلّ هو في تصميم النظام الذي ينفذ العديد من البرامج في الوقت ذاته، وذلك بأن يحوّل المعالج انتباهه الى البرنامج B في الوقت الذي تدخل فيه المعطيات البرنامج A. وسنتناول المعالجة المتوافقة بتفاصيل أكثر في الفصل العاشر. وفي هذه الأثناء دعنا نرى أثرها على تصميم الكمبيوتر الرئيسي.

ان التحكم في الدخل والخرج يشمل وظائف منطقية مثل اختيار الممرّ الذي ستتدفّق عبره المعطيات معدّدة الرموز وحاصية عناوين الذاكرة الرئيسية. ان المعالج هو المصدر الوحيد للمنطق في نظام الميكروكمبيوتر؛ وهكذا، يجب أن يكون المعالج مرتبطاً بكل عملية دخل/خرج. وفي الوقت الذي يتحكّم فيه المعالج بعمليات الدخل والخرج، فانه لا يستطيع تنفيذ تعليمات برنامج تطبيقي، ولكن طبيعة نظام الميكروكمبيوتر تجعل هذه المشكلة مشكلة ثانوية.

مع ذلك، فانه من المعقول بالنسبة الى كمبيوتر يستخدمه العديد من المستعملين، تنفيذ العديد من البرامج المتوافقة. ان دورة آلة الكمبيوتر الرئيسي الأساسية مطابقة تماماً للدورة الرئيسية لآلة الميكروكمبيوتر - إذ ان معالجه أيضاً يستحضر وينفذ تعليمة واحدة في كل مرة. كيف يمكن إذن لمثل هذه الآلة أن تنفّذ برنامجين أو أكثر في وقت واحد؟ إن المفتاح هو في تحرير المعالج من مسؤولية الدخل والخرج. ففي معظم الأنظمة الكبيرة تُعيّن قنوات للقيام بمهمة التحكم في الدخل والخرج (شكل 5.12). والقناة هي ميكروكمبيوتر أو مينيكمبيوتر له معالجه الخاص به؛ ويستطيع أن ينجز المهام المنطقية بالتوازي مع معالج الكمبيوتر الرئيسي، وهكذا يحرّر المعالج الرئيسي للقيام بأعمال أخرى.

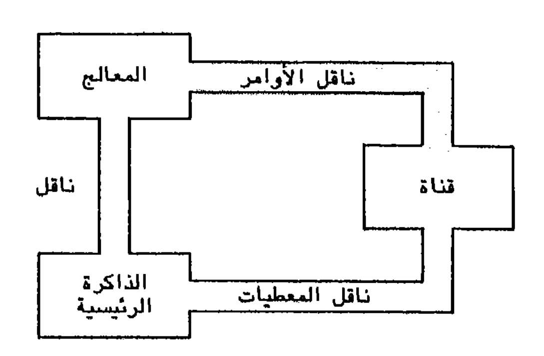
الشكل 5.12 في كمبيوتر رئيسي، مهام مستقلة عن الجهاز، تكلّف قناة بالقيام بها، ومهام معتمدة على الجهاز تكلف وحدة تحكم في الدخل والخرج للقيام بها.



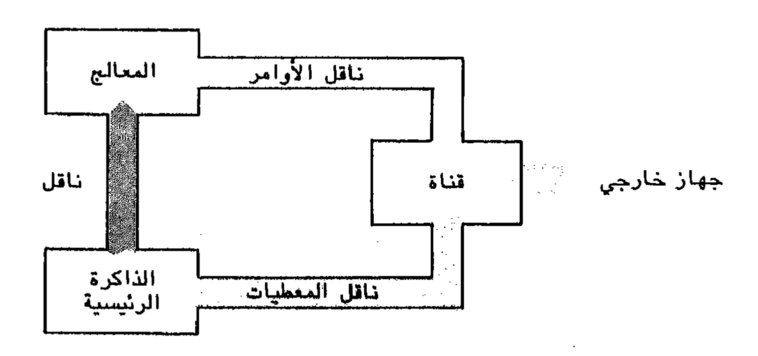
ان بعض مهام الدخل/الخرج تعتمد على الجهاز. فعلى سبيل المثال، إن التحكّم في حركة ذراع نيل هي مشكلة قرص، في حين ان تحويل الرموز الى نمط نقطي هو أمر فريد بالنسبة الى طابعة صفيفية نقطية. أمّا المهام الأخرى مثل اختيار ممر المعطيات وعد الرموز وإحصاء عناوين الذاكرة الرئيسية فهي مشتركة بين كل عمليات الدخل/الخرج بصرف النظر عن الجهاز المحيطي الذي يقوم بالعملية. فتتولى القناة معالجة هذه المهام التي لا تعتمد على الجهاز، في حين تكلّف وحدة تحكم في الدخل/الخرج بالقيام بالمهام المعتمدة على الجهاز (شكل 5.12). ولكل جهاز وحدة تحكمه الخاصة به.

ان القناة تحرّك المعطيات بين الذاكرة الرئيسية وجهاز محيطي. ويعالج معالج الكمبيوتر المعطيات في الذاكرة الرئيسية. ولكن السماح لقناة ومعالج أن ينالا من الذاكرة في آن واحد، هو شيء غير قابل للعمل في نظام الميكروكمبيوتر، لأن تصميم الناقل الأحادي في الميكروكمبيوتر يوفر ممراً واحداً فقط للمعطيات المادية. ان العمليات المتزامنة تتطلّب ممرات معطيات مستقلة، ولذلك فان معظم الكمبيوترات الرئيسية تستخدم **تصميم الناقلات المتعدّدة** (شكل 5.13).

الشكل 5.13 كثير من الكمبيوترات الرئيسية تستخدم تصميم الناقلات المتعددة.

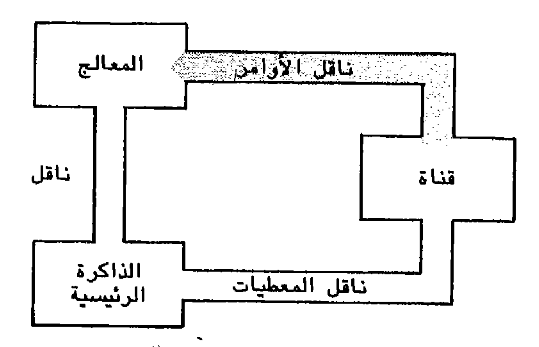


أ. يبدأ المعالج الرئيسي عملية الدخل/الخرج بإرسال إشارة رقمية الى القناة.



ب. تتولى القناة القيام بمسؤولية عملية الدخل/الخرج، ويحول المعالج اهتمامه الى برنامج آخر.

الشكل 5.13

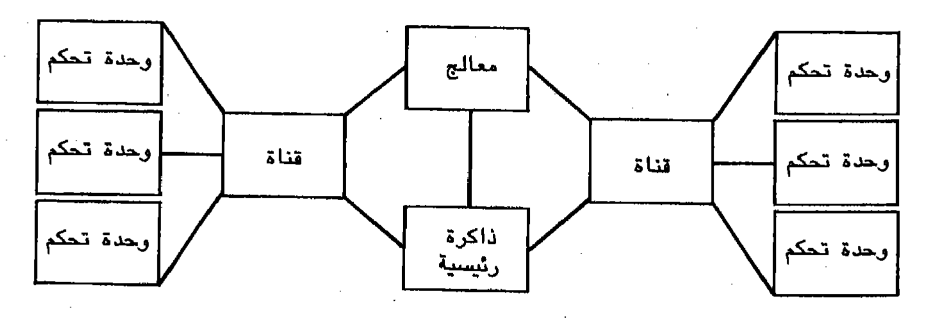


ج. ترسل القناة اشارة انقطاع الى المعالج لتدلّ على نهاية عملية الدخل/والخرج.

دعنا نرى كيف ان عملية دخل/خرج تعتمد على آلة متعدّدة الناقلات. يبدأ المعالج الرئيسي عملية دخل/خرج بإرسال إشارة الكترونية عبر ناقل الأمر الى القناة (شكل أ5.13)، في الوقت الذي تتحرك فيه معطيات A الى الذاكرة الرئيسية عبر ناقل معطيات القناة (شكل ب5.13)، فإن المعالج، مستخدماً ناقل المعطيات الخاص به، يعالج معطيات البرنامج B. وعندما تنتهي عملية الدخل/الخرج، تخبر القناة المعالج بإرسال إشارة الكترونية تُسمى **انقطاعاً** عبر ناقل أوامر (شكل ج5.13). يستطيع المعالج الآن أن يعود الى البرنامج A.

مادياً، يشبه الكمبيوتر الرئيسي الميكروكمبيوتر. إبدأ بلوحة أم أكبر بكثير تتألف من عدد من الشقوق موصولة بخطوط ناقلات. وتوصل لوحات الدارات التي تحمل المكوّنات الداخلية الرئيسية بالشقوق. وفي الكمبيوتر الرئيسي تحلّ القنوات محل البينيات (شكل 5.14)، ويلحق العديد من الأجهزة الخارجية بكل بينية. وبدلاً من ناقل واحد يصل كل المكونات، فإن لمعظم الكمبيوترات الرئيسية العديد من خطوط الناقلات التي توفر ممرات مستقلة بين المكونات (شكل 5.14)؛ ويسمح هذا بالعمليات المتزامنة التي يقوم بها المعالج والقنوات، والتي تسمح، بدورها، للنظام بأن يتحمّل مستعملين عديدين في آن واحد.

الشكل 5.14 قد يكون للكمبيوتر الرئيسي قنوات عدة، والعديد من وحدات التحكم والأجهزة المحيطية الملحقة بكل واحدة منها.



مرة أخرى، عندما نركّز على الالكترونيات، فإننا نميل الى التغاضي عن الشيء الواضح. ان الكمبيوتر الرئيسي قد يتحمّل العديد من أجهزة الخزن الثانوي والمئات من أجهزة الدخل/الخرج. وإذا كانت قاعدة جهاز واحد لكل شق سارية المفعول، فإن الكمبيوتر الرئيسي يجب أن يكون ضخماً لكي يوفّر المكان الضروري لتوصيل كل هذه الأجهزة المحيطية. وعوضاً عن ذلك، فإن قنوات قليلة فقط توصل مباشرة بالكمبيوتر. ثم توصل أجهزة محيطية بالقنوات. ان هذا التصميم المرن يسمح للكمبيوتر الرئيسي الأساسي نفسه أن يتحمّل نظاماً صغيراً أو نظام قنوات متعددة كبيراً تتبعه المئات من الأجهزة المحيطية.

الخلاصة

بحثنا في الفصول الأربعة الأولى مكوّنات الكمبيوتر الرئيسية. وفي هذا الفصل حوّلنا انتباهنا الى تجميع الكمبيوترات. توصل مكونات الكمبيوتر الداخلية بخطوط ناقلات. وتوصل الأجهزة المحيطية بالكبلات. خطوط الناقلات والكبلات المتوازية تحرّك المعطيات بالتوازي، عدة خوينات في كل مرة. أمّا الكبلات التسلسلية، فإنها تحرّك الخوينات الواحدة بعد الأخرى.

أمّا في معظم الكمبيوترات، تصمّم المكونات الداخلية حول حجم كلمة مشترك. ويؤثر حجم الكلمة على سرعة الكمبيوتر وسعة ذاكرته ودقّته وكلفته. وفي بعض الأحيان يمكن زيادة سعة الذاكرة ودقّتها بالتضحية بسرعة المعالجة.

يُبنى الميكروكمبيوتر حول هيكل معدني يُسمّى اللوحة الأم. وتُضاف الملامح الأخرى بتوصيل لوحات الذاكرة ومختلف اللوحات البينية الى الشقوق المتوافرة. ويحد عدد الشقوق من عدد الأجهزة المحيطية التي يمكن أن تُضاف الى الكمبيوتر. ونموذجياً، يتطلّب كل جهاز محيطي لوحته البينية الخاصة به.

تتحمّل الكمبيوترات الرئيسية، غالباً، العديد من المستعملين في آن واحد. وحتى لا يضيع وقت المعالج الرئيسي في التحكّم في الدخل والخرج، تنقل هذه المسؤولية الى قناة تتصل بالكمبيوتر وتتولى القيام بعدد من الوظائف المستقلة عن الجهاز. ويعهد بهذه المهام المعتمدة على جهاز خارجي، الى وحدات تحكم توصل بالقنوات. ولأن للقناة معالجها الخاص، فإنها تستطيع أن تعمل آنياً مع المعالج الرئيسي. وعندما تكمل قناة عملية الدخل والخرج، تحيط المعالج الرئيسي علماً بذلك بإرسال إشارة الكترونية تُسمّى انقطاعاً.

مصطلحات أساسية

- ☐ التصميم
- ☐ الناقل
- ☐ انقطاع
- ☐ كمبيوتر رئيسي
- ☐ ميكروكمبيوتر
- ☐ مينيكمبيوتر
- ☐ اللوحة الأم
- ☐ تصميم متعدد الناقلات
- ☐ متوازٍ
- ☐ متسلسل
- ☐ ناقل احادي
- ☐ شق
- ☐ كلمة

اختبار ذاتي

1. مكونات الكمبيوتر الداخلية موصولة بـ__________.
   أ. كبلات
   ب. كيانات منطقية
   ج. خطوط ناقلات
   د. ليست موصولة

2. تتدفّق المعطيات فوق الناقل في __________.
   أ. تسلسل
   ب. تواز
   ج. إما تسلسل أو تواز
   د. لا هذا ولا ذاك

3. في خط __________، تنقل الخوينات الواحدة تلو الأخرى.
   أ. متوازٍ
   ب. متزامن
   ج. متسلسل
   د. متعاقب

4. في معظم الكمبيوترات تصمّم المكونات الداخلية حول __________ مشترك.
   أ. ناقل
   ب. دورة آلة
   ج. حجم كلمة
   د. دورة مؤقتة

5. يؤثر حجم الكلمة على __________ الكمبيوتر.
   أ. سرعة
   ب. سعة ذاكرة
   ج. دقة
   د. كل ذلك

6. في نظام الميكروكمبيوتر يمكن التضحية بـ__________ من أجل سعة الذاكرة.
   أ. سرعة معالجة
   ب. حجم الكلمة
   ج. الدقة
   د. كل ذلك

7. تُبنى معظم الميكروكمبيوترات حول هيكل معدني يُسمّى __________.
   أ. كمبيوتر رئيسي
   ب. لوحة أم
   ج. شق
   د. ناقل

8. عدد الأجهزة المحيطية التي يمكن أن توصل بالميكروكمبيوتر تُحدَّد بعدد __________ المتوافرة.
   أ. الشقوق
   ب. خطوط الناقلات
   ج. قنوات
   د. كل ذلك

9. تستعمل معظم الميكروكمبيوترات تصميم __________.
   أ. متعدد الناقل
   ب. أحادي الناقل
   ج. متزامن
   د. قياسي

10. في نظام الميكروكمبيوتر، لكل جهاز محيطي ________ خاص/خاصة به.

أ. قناة
ب. بينية
ج. وحدة تحكم
د. خط ناقل

11. في الميكروكمبيوتر يتحكّم ________ مباشرة في دخل/خرج.

أ. قناة
ب. معالج
ج. ناقل
د. لوحة بينية

12. في كمبيوتر كبير يتحكم/تتحكم ________ في دخل/خرج.

أ. قناة
ب. معالج رئيسي
ج. وحدة تحكم
د. لوحة بينية

13. في كمبيوتر كبير توكّل الوظائف التي ينفرد بها جهاز محيطي لـ________.

أ. وحدة تحكم
ب. لوحة بينية
ج. قناة
د. معالج

14. تكون معظم الكمبيوترات الرئيسية ذات تصميم ________.

أ. أحادي الناقل
ب. متعدّد الناقل
ج. متزامن
د. متداخل

15. تخطر قناة المعالج بأن عملية دخل/خرج انتهت بإرسال ________.

أ. نبضة
ب. أمر
ج. انقطاع
د. لا اتصال بينهما

الاجابات

1.ج 2.ب 3.ج 4.ج 5.د 6.أ 7.ب 8.أ 9.ب 10.ب 11.ب 12.أ 13.أ 14.ب 15.ج.

ربط المفاهيم

1. ميّز بين نقل المعطيات بالتوازي وتسلسلياً.
2. كيف توصل المكوّنات الداخلية للكمبيوترات مادياً بعضها ببعض؟
3. في معظم الكمبيوترات تصمّم كل المكونات الداخلية حول حجم كلمة. لماذا؟
4. إشرح كيف يؤثر حجم كلمة الكمبيوتر على سرعة معالجته، وسعة ذاكرته الرئيسية ودقته.
5. ميّز بين الميكروكمبيوتر والمينيكمبيوتر والكمبيوتر الرئيسي.

6. في وصفنا لتصميم الميكروكمبيوتر استعملنا المصطلحات: «لوحة أم، الشق، والناقل». أوجد الصلة المنطقية بين هذه المصطلحات.
7. ماذا يعني مصطلح «تصميم» عندما يطبّق في الكمبيوتر؟
8. في نظام ميكروكمبيوتر نموذجي لكل دخل/خرج وجهاز خزن ثانوي بينيته الخاصة. لماذا؟
9. ميّز بين تصميم أحادي الناقل ومتعدّد الناقل.
10. ما هو الانقطاع، باختصار؟ ليست لديك حتى الآن معلومات كافية لتعريف هذا المصطلح، ولكن ما هو الانقطاع في رأيك؟ سنعود الى هذا المفهوم في الفصل العاشر.

.6

نظام التشغيل

مفاهيم أساسية

التفاعل البيني لكيانَي
الكمبيوتر المنطقي والمادي

الاتصال مع نظام التشغيل
- لغة الأمر

نظام التحكم بالدخل والخرج

تحميل نظام التشغيل
- التحميل البدئي

بعض أنظمة التشغيل

التفاعل البيني لكيانَي الكمبيوتر المنطقي والمادي

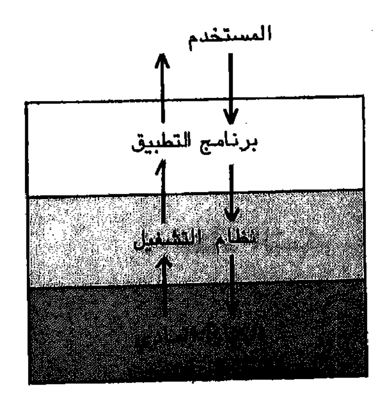
ركّزت الفصول الخمسة الأولى على كيان الكمبيوتر المادي. وقد حان الوقت الآن لتركيز اهتمامنا على الكيان المنطقي.

عندما تفكر في الكيان المنطقي فأنت على الأرجح ستفكر في **برامج التطبيق**. فهي تساند وظائف المستخدم الاخير متيحة المجال للناس لتأدية مهمات مختلفة كممارسة لعبة ما أو كتابة مقالة أو تصميم صفحة بيانية أو تحضير شيكات الدفع. وهناك نوع آخر من الكيان المنطقي ويُسمى **بالكيان المنطقي النظامي** والذي ينفّذ المهمات المخصّصة له سراً. ونموذج مثالي لذلك هو **نظام التشغيل** الموجود في معظم الكمبيوترات. ويتصرّف نظام التشغيل كبينية (شكل 6.1) بحيث يصل الكيان المادي بمجموعة البرامج. وهكذا فعند انتقالنا من الكيان المادي الى الكيان المنطقي يكون من المناسب أن ننظر أولاً في أمر نظام التشغيل.

ماذا يفعل نظام التشغيل تماماً؟ انه ينفذ أساساً عدداً من الوظائف المساندة، فمثلاً، تصوّر برنامج تطبيق مخزوناً في قرص. قبل أن يصبح من الممكن تنفيذ البرنامج، يجب نسخه أولاً في الذاكرة الرئيسية، لأن البرنامج الذي يتحكّم في الكمبيوتر يجب أن يكون موجوداً في الذاكرة الرئيسية. وتتضمّن عملية نسخ برنامج من قرص ما الى الذاكرة منطقاً جديراً بالاعتبار. ومصدر منطق الكمبيوتر هذا هو الكيان المنطقي. وهكذا فانه إذا كان من المطلوب تحميل برنامج التطبيق، يجب أن يكون هناك برنامج في الذاكرة للتحكم في عملية التحميل. وذلك البرنامج ما هو إلاّ نظام التشغيل.

أمّا برامج التحميل فما هي إلّا وظيفة مساندة واحدة من بين الوظائف العديدة المساندة لنظام التشغيل، الذي هو جوهرياً مجموعة من وحدات تركيبية للكيان المنطقي تعزل المستخدم عن الكيان المادي، لتسهّل استخدام النظام. ولنبحث الآن في الوظائف الأساسية لنظام تشغيل نموذجي.

الشكل 6.1 يتصرف نظام التشغيل كبينية ما بين الكيان المادي والكيانات المنطقية التطبيقية.

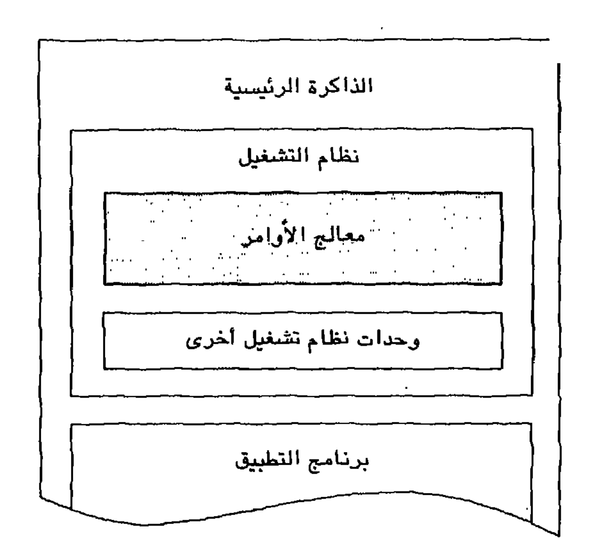


الاتصال مع نظام التشغيل

الكمبيوترات ليست ذكية. فقبل أن يتمكن نظام التشغيل من تأدية واحدة من وظائفه المساندة، فان على الشخص الذي يستعمل الكمبيوتر أن يعلم الكمبيوتر ما هو متوجّب عليه فعله. المستخدم، وهو يشبه الى حدّ بعيد ضابطاً عسكرياً، يعطي الأوامر، فيتجاوب نظام التشغيل وكأنه الرقيب، فيجمع الموارد الهامّة وينفذ كل أمر. تُسمى وحدة نظام التشغيل التي تقبل وتفسّر وتنفذ الأوامر **معالج الأوامر** (شكل 6.2).

يشتمل معالج الأوامر على مجموعة وحدات، كل منها ينفذ مهمة واحدة (شكل 6.3). على سبيل المثال، تحتوي وحدة واحدة على التعليمات التي توجّه الكمبيوتر خلال عملية نسخ برنامج من قرص ما الى الذاكرة الرئيسية وتحميله فيها. وتحتوي وحدة أخرى على التعليمات التي تقوم بنقل تحكم الكمبيوتر الى ذلك البرنامج.

الشكل 6.2 تُسمّى وحدة نظام التشغيل التي تقبل وتفسّر وتنفّذ الأوامر بمعالج الأوامر.



الشكل 6.3 يتركّب معالج الأوامر من عدد من الوحدات البرامجية، كل منها تنجز وظيفة منطقية واحدة.

الذاكرة الرئيسية

نظام التشغيل

معالج الأوامر

أمر التفسير	برنامج التنفيذ	برنامج التحميل	تكوين ملف
نسق القرص	جدولة الفهرس	فحص القرص	وظائف أخرى

مناهج نظام التشغيل الأخرى

برنامج التطبيق

لغة الأمر

يتمّ اتصال المبرمج مع معالج الأوامر بواسطة **لغة الأمر**. وهناك عامة كلمة أمر واحدة بسيطة لكل وظيفة رئيسية - LOAD (حمل برنامج من قرص)، RUN (نفّذ البرنامج المخزون في الذاكرة الرئيسية)، FORMAT (نسق قرص)، DISKCOPY (نسخ قرص) وهكذا دواليك. يطبع المستخدم الجالس أمام لوحة المفاتيح أمراً ما. تتدفق السمات من لوحة المفاتيح عبر لوحة بينية فعبر ناقل فالى الذاكرة الرئيسية. وفور وصولها الى الذاكرة، يفسر معالج الأوامر الأمر ويعطي التحكم الى الوحدة الوظائفية المناسبة.

على سبيل المثال، ادرس مهمة تحميل وتنفيذ برنامج. عند بدء العملية، (مثلاً) طلب إدخال، يتم عرض أ > |ب على الشاشة. جواباً على ذلك يطبع المستخدم

LOAD MYPGM

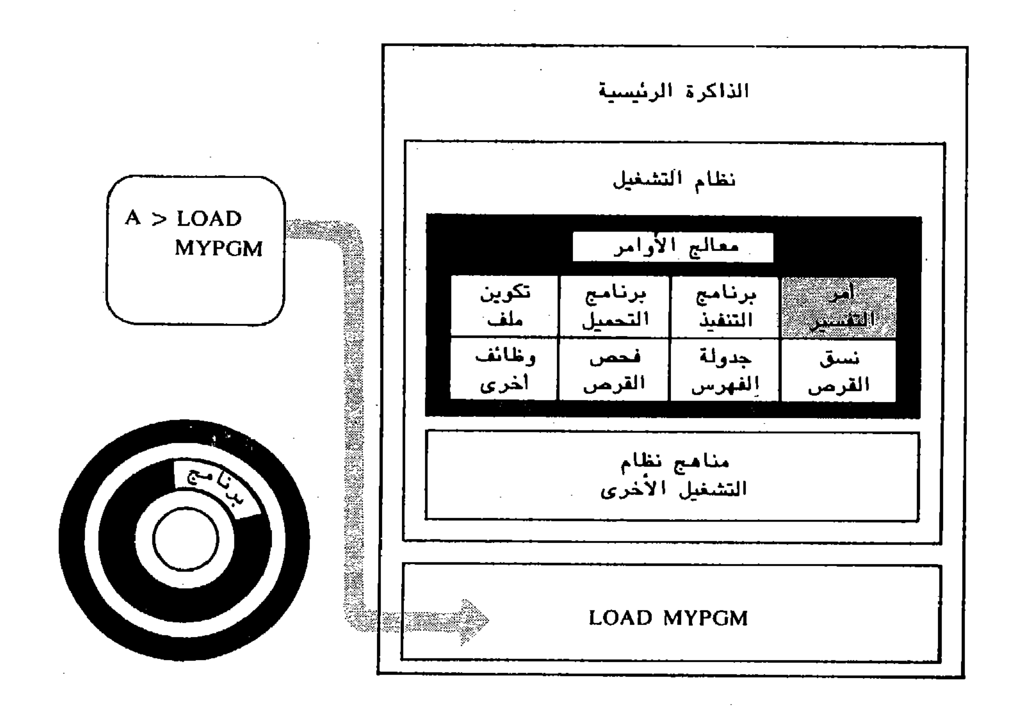
ويضغط على مفتاح الدخول (شكل 6.41). يصل الأمر الى الذاكرة الرئيسية. فيقوم معالج الأوامر بتقييمه، وبملاحظته لأمر LOAD، ينقل التحكم الى وحدة برنامج التحميل (شكل 6.4)، التي تقوم بقراءة البرنامج المطلوب من القرص الى الذاكرة. وما أن يتمّ تحميل

البرنامج، حتى يسترجع معالج الأوامر التحكّم ويعرض طلب ادخال آخر ثم ينتظر الأمر التالي (شكل ج6.4).
لنفترض ان الأمر التالي هو:

RUN

هذا الأمر يدفع بنظام التشغيل الى تنفيذ البرنامج التطبيقي المخزون في الذاكرة الرئيسية. ومتبعاً أمر RUN، يعطي معالج الأوامر التحكم الى الوحدة التي تبدأ برنامج التطبيق (الشكل د6.4). وعندما يتم تنفيذ برنامج التطبيق، يعيد التحكم الى معالج الأوامر الذي يعرض طلب إدخال وينتظر الأمر التالي.

الشكل 6.4 نظام التشغيل هو المسؤول عن تحميل برنامج تطبيق واعطائه التحكم.



1. استجابة لطلب إدخال من نظام التشغيل، يقوم مستخدم بطبع أمر التحميل. ثم يفسّر معالج الأوامر الأمر.

الشكل 6.4

الذاكرة الرئيسية

نظام التشغيل

معالج الأوامر

تكوين ملف	برنامج التحميل	برنامج التنفيذ	امر التفسير
وظائف أخرى	فحص القرص	جدولة الفهرس	نسق القرص

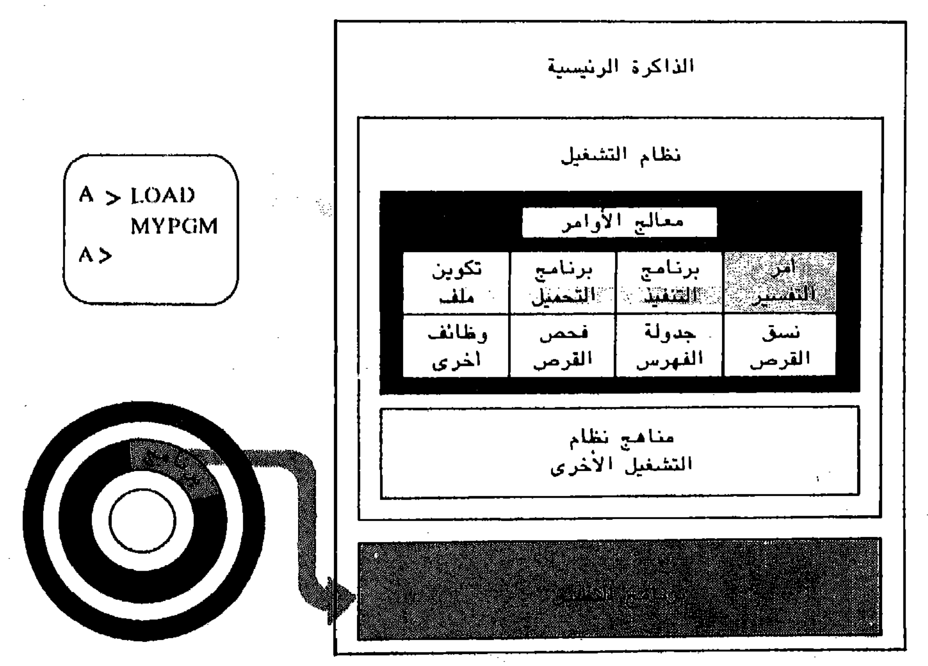
مناهج نظام التشغيل الأخرى

LOAD MYPGM

A > LOAD MYPGM

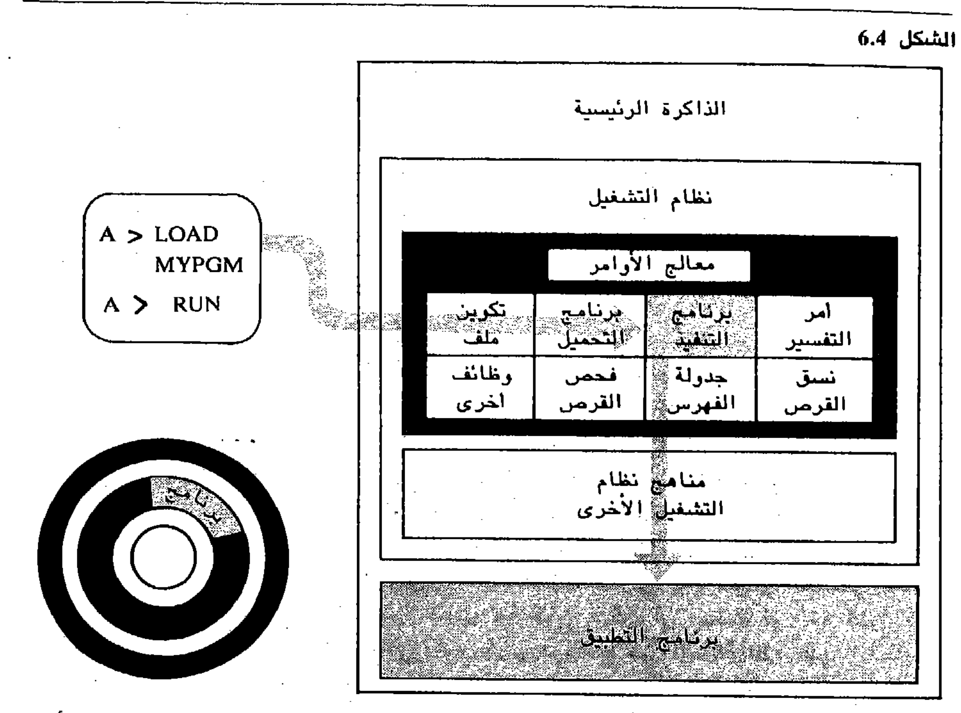
برنامج

ب. يطلب الأمر تحميل برنامج. وهكذا تحصل وحدة برنامج التحميل لمعالج الأمر على التحكم.



ج. بعد أن يتمّ تحميل البرنامج، يحصل معالج الأوامر على التحكم ويعرض طلب الادخال الخاص به ثم ينتظر الأمر التالي.

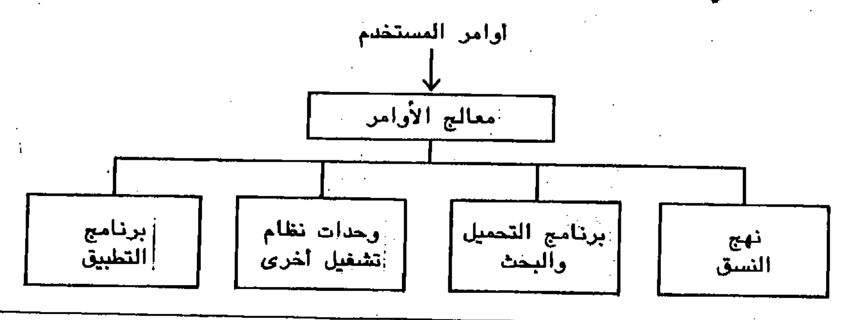
الشكل 6.4



د. يطلب الأمر التالي من نظام التشغيل تنفيذ البرنامج الموجود في الذاكرة. وهكذا يعطي معالج الأوامر التحكم الى الوحدة التي تبدأ برنامج التطبيق.

في معظم الميكروكمبيوترات يكون معالج الأوامر هو وحدة التحكم الرئيسية لنظام التشغيل (شكل 6.5)، فهو يتقبّل الأوامر ويفسّرها ويقرر أي وحدات فرعية يحتاج اليها لتنفيذ الأوامر. تتصل هذه الوحدات مباشرة مع الأجهزة. وعندما تنجز مهمتها، تعيد التحكّم الى معالج الأوامر الذي يقوم بعرض طلب الادخال وينتظر الأمر التالي. يُقال عن هذه الأنظمة بأنها تحث بأمر.

الشكل 6.5 معظم أنظمة التشغيل في الميكروكمبيوترات تحث بأمر. يصدر مستخدم ما امراً. واستجابة للأمر يقوم معالج الأوامر بتحديد ما يتوجّب فعله. وبعد تنفيذ الأمر، ينتظر النظام الأمر التالي.



نظام التحكم بالدخل والخرج

تقوم إحدى الوحدات التي تمّ وصفها في القسم السابق بالبحث عن برنامج وتحميله. وعند شرحها، قمنا بشرح عدد من التفاصيل. هناك تدخّل للمنطق في عملية البحث عن برنامج وتحميله.

على سبيل المثال، افترض ان البرنامج قد تمّ خزنه على قرص. ان في مدوار القرص نسبة ذكاء طفيفة أو لا ذكاء على الاطلاق، فهذه النسبة محصورة في تنفيذ بعض العمليات الأولية والتي تتضمّن الآتي:

1. البحث عن سكة مختارة.
2. قراءة قطاع مختار من السكة.
3. كتابة قطاع مختار الى السكة.

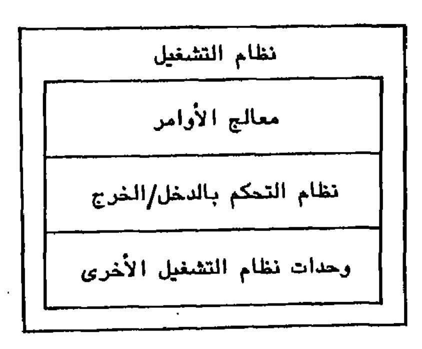
وهذا كل ما يمكن أن يؤدّيه مدوار قرص. إن الطريقة الوحيدة لقراءة برنامج من قرص ما الى الذاكرة الرئيسية تكون في إرسال سلسلة من **الأوامر الأولية** الى المدوار تطلب منه القيام بالبحث عن كل قطاع يحتفظ بجزء من البرنامج وقراءته واحداً تلو الآخر. انتبه الى انه يتوجّب إعلام مدوار القرص بالدقة أين يجب أن توضَع آلية القراءة/الكتابة، وعن القطاعات بالذات التي عليه قراءتها.

من أين تأتي هذه الأوامر؟ قد يكون المستخدم البشري أو الكيان المنطقي مصدر الذكاء والمنطق في جهاز الكمبيوتر. تخيّل ما كان يمكن أن يحدث لو انه كان عليك أن تترابط مع النظام بمستوى بدائي. إذا خزّن برنامج في السكة 20، على القطاعين 8 و9، فعليك أن تطلب من الجهاز بأن:

SEEK 20
READ 8
SEEK 20
READ 9

وبالطبع أنت مسؤول أن تتذكر بأن البرنامج قد تمّ خزنه في السكة 20، على القطاعين 8 و9. وماذا يحدث فيما لو نسيت ان هناك احتمالاً بأن لا ترى برنامجك مرة ثانية.

الشكل 6.6 معظم انظمة التشغيل تحتوي أيضاً على نظام تحكم دخل/خرج. إن نظام التحكم بالدخل والخرج هو الوحدة التي تترابط مباشرة مع الأجهزة المحيطية.



بما ان ما يبتغيه المستخدم هو برنامجاً، فهل حقاً يبدي اهتماماً بمكان الخزن المادي لذلك البرنامج؟ على الأرجح قد لا يفعل ذلك. إن المستخدم النموذجي بكل بساطة يريد برنامجاً. أمّا تفاصيل الكيان المادي الأولية المترافقة مع عملية البحث عن البرنامج وتحميله، يجب أن تكون من اهتمامات الكمبيوتر. هنا يأتي دور نظام التشغيل. ومعظم أنظمة التشغيل تحتوي على **نظام تحكم بالدخل/الخرج** IOCS (شكل 6.6) الذي يولد الأوامر الأولية الهامّة.

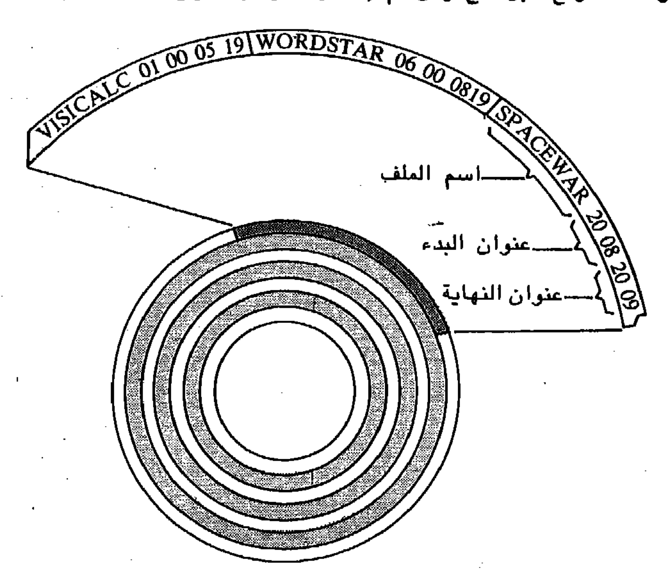
تصوّر ان برنامجاً يُسمّى SPACEWAR قد تمّ خزنه في السكة 20، القطاع 8. أي ستجد من الأسهل تذكره؟ السكة 20، القطاع 8، أو SPACEWAR؟ غالبية الناس تتخيّل أسماء للبرامج. إن هدف نظام التشغيل هو عزل المستخدم عن الكيان المادي - ليصبح الكيان المادي أسهل استعمالاً. وبما ان الناس يجدون سهولة في تذكّر البرامج بأسمائها، يصبح هناك معنى لتصميم نظام التحكم بالدخل/الخرج بشكل يقبل فيه الاسم ويترجمه الى أوامر مادية كيف يتم هذا؟

تذكر ما ورد في الفصل 4 بأن البرامج المختلفة المخزونة على قرص قد تمّ ادراجها في فهرس القرص (شكل 6.7). هذا الفهرس هو السر الذي يؤدي الى نيل البرامج بذكر الاسم. على سبيل المثال، استجابة لأمر

LOAD SPACEWAR

يقوم معالج الأوامر بنقل التحكم الى وحدة تحميل البرنامج. وتقوم تلك الوحدة، بدورها، باستدعاء نظام التحكم بالدخل/الخرج الذي يقرأ الفهرس. وفور وصول الفهرس الى الذاكرة الرئيسية، يصبح بامكان نظام التحكم بالدخل والخرج البحث عن البرنامج. يُعرف

الشكل 6.7 الفهرس الذي يوجد على كل قرص هو السر الذي يؤدّي الى نيل البرامج بأسمائها. كل برنامج على القرص يمكن التعرّف عليه بالاسم، ويحدّد موقعه. ولأجل تحميل برنامج، يقوم نظام التشغيل بقراءة الفهرس والبحث فيه عن الاسم المرغوب اطلاقه على البرنامج وكشف موقع البرنامج ومن ثم إصدار الأوامر الأولية الهامّة لأجل قراءته.



كل برنامج على القرص باسم معين. لاحظ إن إدخال SPACEWAR هو الادخال الثالث. تلي خطوة تسمية البرنامج خطوة تحديد موقعه المادي (بمعنى آخر، السكة والقطاع اللذان يحتفظان بالتعليمات الأولى). باستخدام هذه التعليمة، يصدر نظام التحكم بالدخل والخرج أمرَي القراءة والبحث الضروريين، ويتم نسخ البرنامج في الذاكرة.

هذه هي كل الخطوات المطلوبة جوهرياً لتحضير برنامج ما. عندما يُكتب برنامج ما في قرص، يتم تسجيل اسمه وموقعه المادي في الفهرس. ولاستعادة البرنامج، يقرأ الفهرس ويبحث فيه عن الاسم ثم يستخرج موقع البرنامج المادي من الفهرس وبعد ذلك تصدر أوامر دخل/خرج الضرورية.

لقد ارتكز مثالنا الى عملية نيل القرص. إن نظام التحكم بالدخل والخرج مسؤول ايضاً عن الترابط مع أجهزة النظام المحيطية الأخرى. وكل جهاز مادي يتمّ التحكم به بواسطة مجموعة أوامره الأولية الفريدة. (وهذا، صدفياً هو سبب آخر يشير الى حاجة كل جهاز دخل أو خرج الى بينيّته أو وحدة تحكمه الخاصة به). ان برامج التطبيق تصدر طلبات عامة لأجل البدء في الدخل أو الخرج. ويقبل نظام التحكم بالدخل/الخرج هذه الطلبات العامة ويولد الأوامر الأولية التي يحتاج اليها للتحكم بجهاز محيطي معين.

وعلى أي حال، فان إنشاء ترابط مع جهاز خارجي يشتمل على أكثر من عملية توليد أوامر أولية فقط. على سبيل المثال، كلما ترابط مكونا كيان مادي (مثل كمبيوتر ومدوار قرص) مع بعضهما البعض، يجب تزامن إشاراتهما الالكترونية بدقة. تشتمل عملية التزامن على تبادل مجموعة إشارات سبق تقديرها تُسمّى «بروتوكول». ان عملية بدء أو فحص إشارات البروتوكول هي عملية مملّة يجري عادة تكليف نظام التشغيل بها.

وعلى العموم، فان هناك عدداً من التفاصيل تترافق مع خطوات البدء والنهاية والتحكم بأية عملية دخل أو خرج. وبدلاً من استخراج نسخ مطابقة للمنطق لتنفيذ هذه الوظائف في كل برنامج تطبيق، فانه من الأجدى تطبيقها مرة واحدة، في نظام التحكم بالدخل والخرج لنظام التشغيل، والسماح لبرامج التطبيق بنيل الأجهزة المحيطية من خلال هذه الوحدة. ومرة أخرى نجد ان نظام التشغيل هو بينية كيان مادي ومنطقي.

تحميل نظام التشغيل

يبدأ تحميل وتنفيذ برنامج بأمر يقرأه ويفسّره نظام التشغيل. وبالطبع يجب أن يكون نظام التشغيل موجوداً في الذاكرة قبل إصدار الأمر. كيف يصل نظام التشغيل الى الذاكرة؟ ففي بعض الأنظمة نجد ان نظام التشغيل مخزون في ذاكرة قراءة فقط. إن ذاكرة قراءة فقط ROM هي ذاكرة دائمة تحتفظ بمحتوياتها حتى ولو فقدت الطاقة. ان نظام تشغيل مرتكزاً على ذاكرة قراءة فقط موجود بشكل دائم.

وعلى أي حال، وفي معظم الكمبيوترات، تتكوّن الذاكرة الرئيسية من ذاكرة نيل عشوائي RAM، إن ذاكرة نيل عشوائي هي ذاكرة متطايرة، تفقد محتوياتها عند انقطاع الطاقة. وهكذا كلما شغل الكمبيوتر، يجب أن يكون نظام التشغيل محملاً. لسوء الحظ لا نستطيع بكل بساطة أن نطبع أمراً، مثل LOAD OS، وندع نظام التشغيل يهتم بتحميل نفسه. لم لا؟ عندما يشغل الكمبيوتر أولاً، تكون الذاكرة الرئيسية فارغة. وإذا لم يكن قد وصل نظام التشغيل الى الذاكرة بعد، فان الذاكرة لا تستطيع قراءة وتفسير وتنفيذ الأمر.

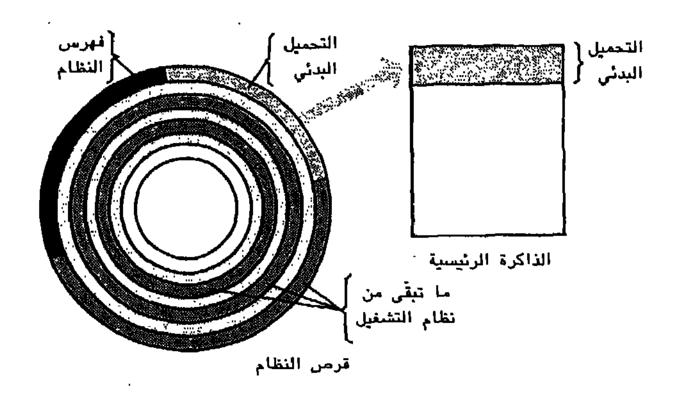
التحميل البدئي

نموذجياً، يخزّن نظام التشغيل على قرص. والفكرة تدعو الى نسخه في الذاكرة. يتحقّق هذا الهدف بواسطة برنامج خاص يُسمّى «**التحميل البدئي**» (شكل 6.8). يخزّن التحميل البدئي عموماً في القطاع الأول (أو في قطاعين) من القرص. لقد صمّم الكيان المادي

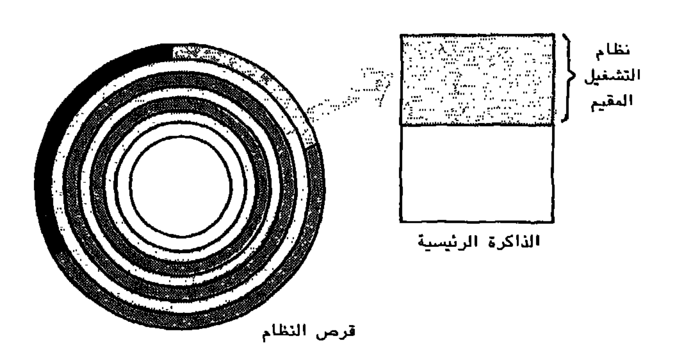
لأجل قراءة هذا القطاع أوتوماتيكياً كلما تمّ وصل التيار الكهربائي الى الجهاز (شكل 6.8أ).

يحتوي التحميل البدئي على تعليمات قليلة فقط إلاّ انها كافية لتقوم بقراءة ما تبقّى من نظام التشغيل على الذاكرة (شكل ب6.8). لاحظ كيف أن التحميل البدئي على ما يبدو يوقف بواسطة تحميله التشغيلي. والآن يستطيع مستخدم ما طبع الأوامر الداعية الى تحميل وتنفيذ برنامج التطبيق.

الشكل 6.8 يتمّ تحميل نظام التشغيل الى الذاكرة الرئيسية بواسطة برنامج خاص يُسمَّى التحميل البدئي.



أ. عندما يشغل الكمبيوتر أولاً، يقوم الكيان المادي بقراءة برنامج التحميل البدئي من القطاعات الأولى على القرص.



ب. يحتوي نهج التحميل البدئي على التعليمات التي تقوم بقراءة ما تبقّى من نظام التشغيل من القرص على الذاكرة.

مثال

دعنا نستخدم مثالاً موجزاً يختصر ما تعلمناه عن نظام التشغيل. عند البداية، نجد الذاكرة الرئيسية فارغة ونجد قرصاً يحتوي على التحميل البدئي وعلى نظام التشغيل وعلى برنامج التطبيق قد تمّ تحميله الى مدوار القرص. وعندما ندير الطاقة الى الكمبيوتر، يتدفّق التحميل البدئي الى الذاكرة الرئيسية وتنفذ تعليماته، ونتيجة لذلك ينسخ نظام التشغيل من القرص.

والآن أصبح التحكم بالكمبيوتر في قبضة وحدة معالج الأوامر في نظام التشغيل. فيعرض طلب ادخال وينتظر عندئذ المستخدم لادخال أمر فيطلب المستخدم من النظام تحميل برنامج ما. عندئذ يقرأ معالج الأوامر الأمر ويفسّره ويسلم عمل التحكم الى وحدة تحميل برنامجه. وبما ان تحميل برنامج ما يستدعي نيل القرص، تقوم هذه الوحدة بدورها بنداء نظام التحكم بالدخل/الخرج. يجد نظام التحكم بالدخل والخرج البرنامج المطلوب وينسخه في الذاكرة الرئيسية. ويعود من ثم التحكم الى معالج الأوامر الذي يعرض طلب إدخال وينتظر الأمر التالي.

ويكون الأمر التالي تنفيذ برنامج RUN. فيقراه معالج الأوامر ويفسّره ثم ينقل التحكم الى الوحدة التي تبدأ برامج التطبيق. يحصل برنامج التطبيق على التحكم فيترابط المستخدم معه. وأخيرا، ينتهي التطبيق فيعود التحكم الى معالج الأوامر الذي يعرض طلب إدخال وينتظر الأمر التالي. يستطيع المستخدم عند هذه النقطة أن يقوم بتحميل برنامج آخر وتنفيذ وظيفة نظامية مثل تنسيق قرص أو يعلن انتهاء النظام.

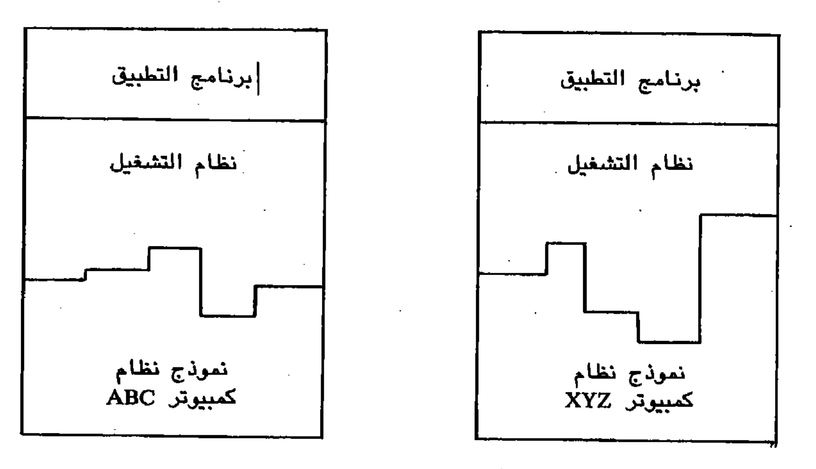
بعض أنظمة التشغيل

ربما يكون أكثر أنظمة تشغيل الميكروكمبيوتر شهرة هو نظام MS/DOS التي طوّرته شركة Microsoft Corporation لحساب IBM-PC. وقد أصبح هذا النظام مقياساً صناعياً.

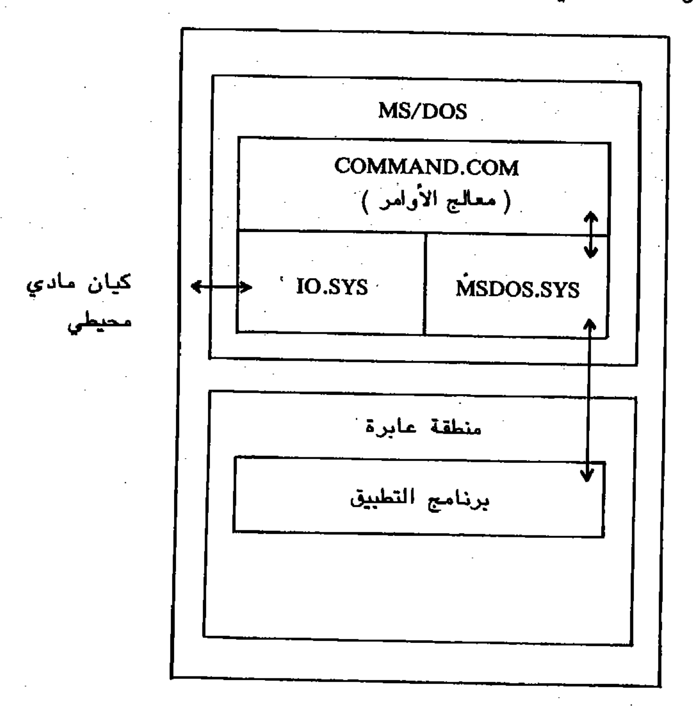
إننا قد نتساءل عن الأهمية التي تُعطى لمستوى نظام تشغيل. على مستوى الكيان المادي، نجد ان الكمبيوترات التي هي من صنع صناعيين مختلفين غالباً ما تكون متضاربة، وبمعنى آخر، فان برنامجاً كُتب لكمبيوتر ما لن يكون صالحاً لكمبيوتر آخر. تذكر على أي حال، بأن نظام التشغيل يكون موجوداً بين الكيان المادي وبرنامج التطبيق. بوجود نظام مشترك في الوسط، يصبح من الممكن لنفس البرنامج أن يُنفّذ على آلتين مختلفتين تماماً (شكل 6.9). وبالطبع قد تكون أقسام نظام التشغيل التي تترابط مع الكيان المادي مختلفة جداً، لكن الكيان المنطقي سيرى بينية مشتركة ومنتظمة.

يتكون MS/DOS من ثلاث وحدات أساسية (شكل 6.10). معالج الأوامر هو COMMAND COM، وتتوزّع وظائف نظام التحكم بالدخل، والخرج ما بين نهجين روتينين. إن أنظمة MSDOS.SYS و IO.SYS و MSDOS.SYS هي أنظمة مستقلة عن الكيان المادي. ومن جهة ثانية، فان نظام IO.SYS يتصل مباشرة مع الكيان المادي، فهو إذا يحتوي على كود يعتمد على الجهاز. يجب أن تختلف الأنواع المختلفة من نظام التشغيل المعدة لكمبيوترات مختلفة عن بعضها البعض فقط في منطق IO.SYS. بالاضافة الى الوحدات الأولية، فان نظام التشغيل يحتوي على عدد من برامج المنفعة العامة.

الشكل 6.9 يعرض نظام التشغيل برنامج التطبيق ببينية منسجمة ومنتظمة.



الشكل 6.10 نظام MS/DOS، ربما يكون أكثر أنظمة تشغيل الكمبيوتر شهرة، وهو يتألف من ثلاث وحدات أساسية.



عندما يجهز نظام MS/DOS، تنسخ أنظمة COMMAND-COM و MSDOS.SYS وIO.SYS الى الذاكرة الرئيسية. وتشتمل المنطقة العابرة (الشكل 6.10) ذاكرة غير مخصصة بنظام التشغيل. يتم قراءة برامج التطبيق ومنافع النظام وبعض الوحدات العابرة والمعطيات في المنطقة العابرة.

وبما ان نظام MS/DOS مستعمل على نطاق واسع، فقد تمّ تطوير مكتبة واسعة من كيانات منطقية تطبيقية لأجله. ان مكتبة الكيان المنطقي تميل الى العمل على ديمومة مرتبة نظام MS/DOS كمقياس لسبب بسيط ألا وهو انه من المعقول جداً أن نشتري كمبيوتر الذي من أجله وجد الكيان المنطقي. كل مورد كمبيوتر هام يقوم فعلياً بدعم نظام MS/DOS، على الأقل كنظام اختياري.

هناك نظام تشغيل آخر شائع الاستعمال هو نظام CP/M وهو عبارة عن برنامج تحكم لميكروكمبيوترات طورته شركة Digital Research, Incorporated لأجل أجهزة ذي 8 خوينات سالفة. ويعد نظام UNIX، الذي طوّرته موسسة American Telephone (AT & T) and Telegraph، يبدو بأن هناك أمل بأن يصبح مقياساً جديداً وخاصة لتطبيقات تتضمّن اتصالات ما بين كمبيوترين أو أكثر. ويكون لدى الكمبيوترات الرئيسية أنظمة تشغيل أكثر تعقيداً والتي بالاضافة الى تصرّفها كبينية كيان مادي وكيان منطقي تقوم بادارة موارد الكمبيوتر. سنقوم ببحث هذه الأنظمة في الفصل العاشر.

الخلاصة

بعد شرح الفرق ما بين الكيانات المنطقية التطبيقية والكيانات المنطقية النظامية تحوّلنا نحو نظام التشغيل الذي هو عبارة عن مجموعة وحدات تتصرف كبينية كيانات مادية وكيانات منطقية. يتصل المستخدمون مع نظام التشغيل بواسطة لغة أوامر. يقبل معالج الأوامر ويفسّرها ثم ينفّذها. وتكون معظم أنظمة تشغيل الميكروكمبيوترات منساقة بأمر.

ان الترابط مع أجهزة الدخل والخرج شيء صعب لأن لكل جهاز أوامره الأولية الذاتية التي تتحكم فيه. يقبل نظام التحكم بالدخل والخرج (IOCS) الطلبات المعمّمة للدخل/الخرج ويولد الأوليات الضرورية. وقد استخدمنا مثالاً لنبين كيف يقبل الـIOCS اسم برنامج ويبحث عنه في فهرس القرص ويجد موضع البرنامج ويقرأ البرنامج الى الذاكرة الرئيسية. ومن وظائف الـIOCS الأخرى هي توليد وتفسير إشارات البروتوكول التي يحتاج اليها لجعل الأجهزة متزامنة قبل أن تبدأ في التواصل.

وبما ان ذاكرة الكمبيوتر الرئيسية تكون متطايرة، يجب تحميل نظام التشغيل كل مرة يتمّ فيها تشغيل الكمبيوتر. ويسمى النهج الذي يقوم بتحميل نظام التشغيل بالتجهيز الذاتي. وهو يخزن نموذجياً في القطاع الأول أو الثاني من القرص ويقرأ التحميل بالكيان المادي. وعند دخوله الذاكرة يقوم بتحميل ما تبقى من نظام التشغيل.

قد يكون أكثر أنظمة تشغيل الميكروكمبيوترات شيوعاً هو نظام MS/DOS. وقد بحثنا باختصار الأجزاء المركبة لـMS/DOS رابطين اياها بالوظائف العامة لنظام تشغيل. إن نظام CP/M هو نظام تشغيل ذي ثمانية خوينات شائع الاستعمال. أمّا نظام UNIX يبشر بأنه سيصبح مقياساً جديداً.

مصطلحات أساسية

- ☐ برنامج التطبيق
- ☐ التحميل البدئي
- ☐ لغة الأمر
- ☐ معالج الأوامر
- ☐ نظام التحكم
- ☐ نظام MS/DOS
- ☐ نظام التشغيل
- ☐ امر اولي
- ☐ طلب إدخال
- ☐ بروتوكول
- ☐ نظام الكيان المنطقي

اختبار ذاتي

1. ______________ يصلح كبينية كيانات مادية وكيانات منطقية.

أ. النظام
ب. برنامج التطبيق
ج. وحدة التحكم
د. نظام التشغيل

2. مصدر منطق الكمبيوتر هو ______________.

أ. الكيان المادي
ب. الكيان المنطقي
ج. المعطيات
د. المستخدم

3. إن وحدة نظام التشغيل التي تقبل الأوامر وتفسّرها وتنفّذها هي ______________

أ. IOCS
ب. معالج الأوامر
ج. التحميل البدئي
د. محمل البرامج

4. يتصل المستخدم مع نظام التشغيل بواسطة ______________.

أ. تعليمات
ب. برنامج
ج. كيان مادي
د. معالج أوامر

5. تكون معظم أنظمة تشغيل الميكروكمبيوترات مساقة ______________.

أ. طلب إدخال
ب. أمر
ج. كيان منطقي
د. كيان مادي

6. يكون مدوار القرص محصوراً ببضعة ______________.

أ. مهمات
ب. وظائف برامج
ج. عمليات أولية
د. عمليات I/O

7. إن وحدة نظام التشغيل التي ترسل أوامرها الى مدوار القرص الأولية هي ______________.

أ. معالج الأوامر
ب. IOCS
ج. التحميل البدئي
د. مداول انقطاع

8. يوجد اسم وموضع كل برنامج تمّ خزنه على قرص في ______________ القرص.

أ. فهرس
ب. اسم الرمزي
ج. بينية
د. أمر أولي

9. يكون ______________ مسؤولاًعن الاتصال مع كل أجهزة الدخل والخرج.

أ. معالج الأوامر
ب. IOCS
ج. التحميل البدئي
د. نظام الكيان المنطقي

10. قبل أن يتمكّن جهازين من الترابط، يجب أن يتزامنا بارسالهما إشارات ________.

أ. تحميل بدئي
ب. انقطاع
ج. أولية
د. بروتوكول

11. تكون ________ دائمة، أمّا ________ فتكون غير مستقرّة.

أ. الذاكرة/الخزن
ب. الخزن/الذاكرة
ج. ROM/RAM
د. RAM/ROM

12. يتمّ تحميل نظام التشغيل بواسطة ________.

أ. معالج الأوامر
ب. تحميل بدئي
ج. محمل البرامج
د. انقطاع

13. إن أكثر أنظمة تشغيل الميكروكمبيوتر شيوعاً على الأرجح، هو نظام ________.

أ. MS/DOS
ب. MVS
ج. VM
د. CPC

14. يبشّر ________ بأن يصبح مقياساً جديداً وخاصة للتطبيقات التي تتضمّن الاتصال ما بين كمبيوترين أو أكثر.

أ. MS/DOS
ب. CP/M
ج. VM
د. UNIX

الاجابات

1.د 2.ب 3.ب 4.د 5.ب 6.ج 7.ب 8.أ 9.ب 10.د 11.ج 12.ب 13.أ 14.د.

ربط المفاهيم

1. ميّز ما بين البرامج التطبيقية والكيانات المنطقية النظامية.
2. يتصرّف نظام تشغيل كبينية ما بين البرامج التطبيقية والكيان المادي. إشرح.
3. إن نظام التشغيل هو عبارة عن مجموعة من وحدات برامجية، يقوم كل منها بتنفيذ وظيفة مساندة واحدة. إشرح.
4. ماذا يفعل معالج الأوامر في نظام التشغيل؟ بيّن الصلة ما بين معالج الأوامر واللغة الأمرية.
5. ماذا يفعل نظام التحكم بالدخل/الخرج في نظام التشغيل؟
6. ما هي الأوامر الأولية؟ ولماذا هي ضرورية؟

7. تخيّل برنامجاً سُمّي بـMYPGM وخزّن على قرص. إشرح باختصار كيف يقوم نظام التشغيل بتحميله الى الذاكرة الرئيسية. ابدأ بأوامر المستخدم.
8. إشرح كيف يتمّ تحميل نظام تحميلاً بدئياً. لماذا يكون التحميل البدئي ضروريا؟
9. ضع مخطّطاً اجمالياً للأجزاء المركبة لنظام تشغيل ميكروكمبيوتر. إشرح باختصار ماذا يفعل كل مكّون؟
10. ضع مخطّطاً اجمالياً للأجزاء المركبة لنظام MS/DOS. قارن مكوّناته بالمكوّنات الموجودة في الرسم التخطيطي للتمرين 9.

7.

الكيان المنطقي (البرامجيات) التطبيقي

مفاهيم أساسية

ما هو الكيان المنطقي؟

لغات البرمجة
- ☐ المؤولات
- ☐ المصرفات والمفسّرات
- ☐ اللغات غير الاجرائية

المكتبات

عملية تطوير البرامج
- ☐ تحديد المشكلة
- ☐ التخطيط
- ☐ كتابة البرنامج
- ☐ كشف الخطأ وتصحيحه والتوثيق
- ☐ الصيانة

إعداد برامجك الخاصة

ما هو الكيان المنطقي (البرامجيات)؟

عرفنا في الفصل 6 أن نظام التشغيل هو مجموعة من وحدات الكيان المنطقي تقوم بوظائف مساندة وتعزل المستخدم فعلاً عن الكيانات المادية. وننتقل في هذا الفصل إلى بحث أكثر شمولية للكيان المنطقي. وليست غايتنا أن نعلمك كيف تعد برنامجاً، فانت لن تتعلم من خلال قراءة فصل واحد في كتاب تمهيدي، وإنما سنحاول عوضاً عن ذلك أن نزيل بعض الغموض الذي يلفّ الكيان المنطقي في أغلب الأحيان من خلال إعطائك صورة عن ماهيّة البرنامج على مستوى الآلة، ومن ثم شرح العملية التي يتبعها المبرمج باختصار عند إعداده لبرنامج ما.

لنبدأ بالتعريف. **البرنامج** هو سلسلة من **التعليمات** التي توجّه الكمبيوتر خلال معالجة، وكل تعليمة تأمر الآلة بتأدية احدى وظائفها الأساسية: إطرح، إجمع، أضرب، أقسم، قارن، إنسخ، أطلب دخلاً، أو أطلب خرجاً. وقد تعلمنا في الفصل 2 أن المعالج يستحضر تعليمة واحدة وينفّذها في كل دورة للآلة. وتحتوي التعليمة النموذجية (الشكل 7.1) على كود عملية يحدّد الوظيفة المطلوب تأديتها، وسلسلة من المعاملات التي تحدّد مواقع الذاكرة أو المراصف التي تحفظ المعطيات التي ستجري معالجتها. على سبيل المثال تأمر التعليمة

(ADD) 3,4

كمبيوتراً افتراضياً بجمع المرصفين 3 و4.

الشكل 7.1 تتألف التعليمة من كود عملية ومعامل واحد او اكثر. ويأمر كود العملية الكمبيوتر بما يجب عمله، بينما يحدّد المعامل أو المعاملات عناوين المعطيات المفروض معالجتها.

كود العملية	المعاملات
إجمع (ADD)	3,4

الشكل 7.2 يجب أن يكون البرنامج موجوداً في شكل ثنائي لأن ذاكرة الكمبيوتر الرئيسية تخزّن الخوينات. وهذه التعليمات الأربع ضرورية لجمع عددين على كمبيوتر آي بي أم رئيسي.

01011000001100001100000000000000
01011000010000001100000000000100
0001101000110100
01010000001100001100000000001000

وحتى تلك العمليات المنطقية البسيطة تستدعي عدة تعليمات لأن مجموعة تعليمات الكمبيوتر محدودة للغاية. على سبيل المثال، لنتصوّر وجود قيمتَي معطيات مخزونتين في الذاكرة الرئيسية. لكي يتمّ جمعها على كثير من الكمبيوترات، تحمل (أو تنسخ) القيمتان أولاً داخل مرصفين ثم يجمع المرصفان، ومن ثم يُعاد خزن (أو نسخ) الاجابة في الذاكرة الرئيسية. يشكّل ذلك كله أربع تعليمات: حمل، حمل، إجمع وخزّن. وإذا كان جمع عددين يتطلب أربع تعليمات فتخيّل عدد التعليمات في برنامج كامل.

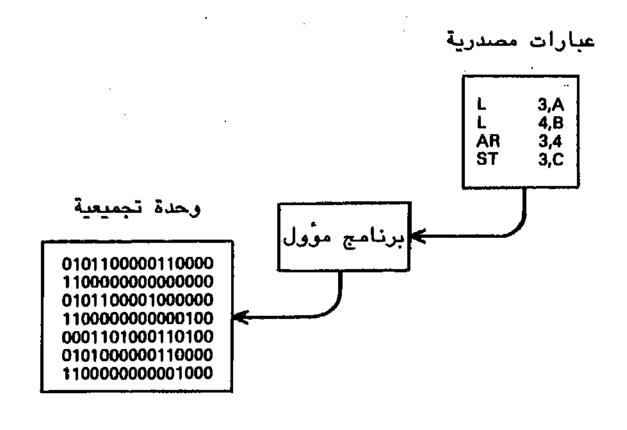
ويتمّ التحكّم بالكمبيوتر بواسطة برنامج مخزون في ذاكرته الرئيسية. ويجب أن يكون البرنامج موجوداً في شكل ثنائي لأن الذاكرة الرئيسية تخزّن الخوينات. ويظهر الشكل 7.2 التعليمات الثنائية على مستوى الآلة المطلوبة لتحميل عددين في مراصف وجمعهما وخزن الاجابة في الذاكرة☆ ولو كان على المبرمجين أن يكتبوا **بلغة الآلة** لكان هناك عدد قليل جداً منهم.

لغات البرمجة

المؤولات (المترجمات الجامعة)

من الخيارات المُتاحة كتابة التعليمات **بلغة التأويل**، على سبيل المثال، يظهر الشكل 7.3 الكيفية التي يمكن أن يجمع بها عددان في مؤول كمبيوتر آي بي أم رئيسي، حيث يكتب المبرمج تعليمة واحدة (مساعدة للذاكرة) مختصرة لكل تعليمة على مستوى الآلة. فإن تذكر الحرفين AR (إختصار لـAdd registers (إجمع المراصف)) أسهل بكثير من تذكّر كود العملية الثنائي المقابل: 00011010، وتذكر الحرف L (إختصار لكلمة Load (حمل)) أسهل بكثير من تذكّر 01011000. وتستخدم المعاملات أسماء رمزية مثل A وB وC بدلاً من الأعداد لتمثيل عناوين الذاكرة الرئيسية، وهذا ما يبسط الكود أيضاً.

الشكل 7.3 يقرأ البرنامج المؤول عبارات المبرمج المصدرية المختصرة، ويترجم كل منها الى تعليمة واحدة على مستوى الآلة، ومن ثم يربط بينها ليشكّل وحدة تجميعية.



☆. تصلح التعليمات الثنائية على كمبيوتر آي بي أم 3083 الرئيسي وكمبيوترات اخرى في عائلة آي بي أم سيستم/370.

لسوء الحظ ليس هناك كمبيوترات تستطيع أن تنفّذ مباشرة تعليمات لغة مؤول (مترجم جامع) ولربما بسطت كتابة الكودات المختصرة عمل المبرمج، لكن الكمبيوترات ما تزال آلات ثنائية وتتطلّب تعليمات ثنائية، وبالتالي كانت الترجمة ضرورية. فالبرنامج المؤول (شكل 7.3) يقرأ **الكود المصدري** الذي يخص المبرمج ويترجم العبارات المصدرية الى عبارات ثنائية وينتج **وحدة تجميعية**. ويمكن تحميل الوحدة التجميعية في الذاكرة وتنفيذها لأنها نسخة على مستوى الآلة من كود المبرمج.

ويكتب مبرمج لغة التأويل تعليمة واحدة مختصرة لكل تعليمة على مستوى الآلة. وبسبب وجود علاقة «واحد لواحد» بين اللغة والآلة، فإن المؤولات تعتمد على الآلة، ولذلك فإن برنامجاً كُتب لنموذج معين من الكمبيوتر لا يصلح للتنفيذ في كمبيوتر آخر. تولد لغة التأويل أكثر البرامج كفاية على آلة معيّنة. وهكذا فهي غالباً تستخدم لكتابة أنظمة التشغيل وغيرها من أنظمة الكيان المنطقي. ومع ذلك فإن التبعية للآلة هي الثمن الباهظ الذي يُدفع لقاء الفعالية عندما تصل الأمور الى البرامج التطبيقية، ولذلك نادراً ما تكتب البرامج التطبيقية بلغة التأويل.

المصرفات والمفسّرات

يحتاج الكمبيوتر الى أربع تعليمات على مستوى الآلة لجمع عددين، لأنه هذه هي الطريقة التي يعمل بها الكمبيوتر. ولا ينبغي على البشر ان يفكروا كالكمبيوترات. فلماذا لا نسمح بكل بساطة للمبرمج بأن يشير إلى عملية الجمع ويفترض التعليمات الأخرى؟ فمثلاً إن إحدى الطرق التي تمثّل فيها عملية الجمع هو التعبير الجبري:

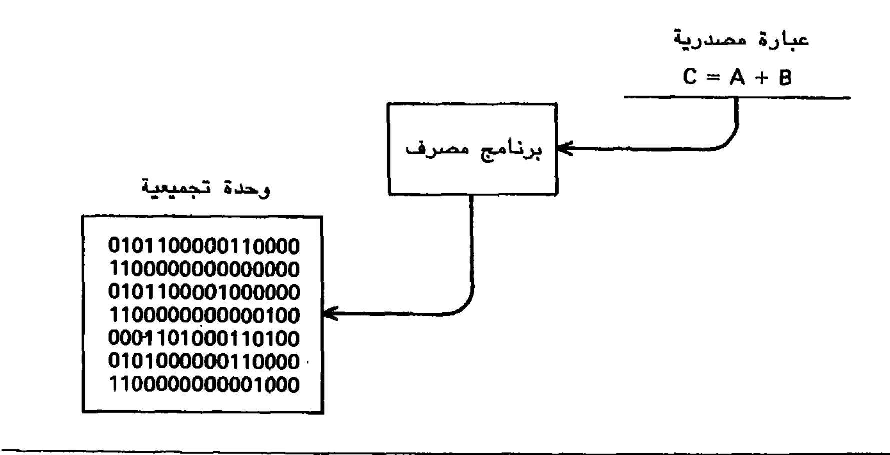
$$C = A + B$$

ولماذا لا نسمح للمبرمج بكتابة عبارات في شكل مشابه للتعبير الجبري، وبقراءة تلك العبارات المصدرية في برنامج، ونترك للبرنامج أن يولد الكود اللازم على مستوى الآلة (شكل 7.4)؟ هذا هو بالضبط ما يحدث مع المصرف. قارن التعليمات الثنائية في الشكلين 7.3 و7.4 فتجد انها متطابقة.

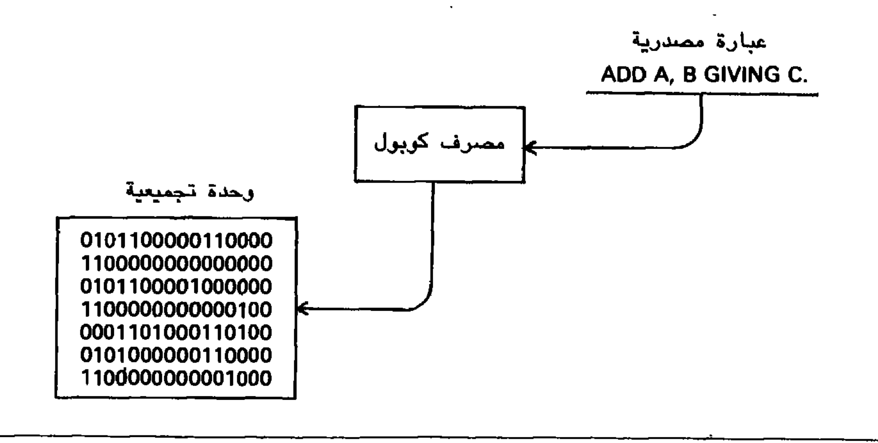
والواقع أن العديد من لغات التصريف، وبينها الفورتران (FORTRAN) والبيسيك (BASIC) وباسكال (PASCAL) وبي آل/1 (PL/1) والغول (ALGOL) تستند إلى الأسس الجبرية. وتستخدم لغة كوبول (COBOL)، وهي أكثر اللغات الموجّهة للتعبير عن الأعمال رواجاً، عبارات تشبه الجمل الانكليزية المختصرة (شكل 7.5). على أنه يُلاحَظ أن الهدف يظل واحداً أياً كانت اللغة التي تستخدم. ويكتب المبرمج الكود المصدري، ويقبل البرنامج المؤول الكود المصدري المختصر، ويولد وحدة تجميعية على مستوى الآلة. ويقبل مصرف الفورتران كود الفورتران المصدري، ويولد وحدة تجميعية على مستوى الآلة، وكذلك الأمر يقبل مصرف الكوبول كود الكوبول المصدري ويولد وحدة تجميعية على مستوى الآلة.

ما هو الفرق بين المؤول والمصرف؟ مع المؤول تحوّل كل عبارة مصدرية إلى تعليمة واحدة على مستوى الآلة، أمّا مع المصرف فيمكن أن تحوّل عبارة مصدرية معيّنة إلى أي عدد من التعليمات على مستوى الآلة.

الشكل 7.4 يقرأ المصرف عبارات المبرمج المصدرية ويترجم كل منها إلى تعليمة واحدة أو أكثر على مستوى الآلة، ومن ثم يربطها معاً ليشكّل وحدة تجميعية.



الشكل 7.5 أكثر اللغات الموجّهة في مجال الأعمال رواجاً هي الكوبول. وكما هو الأمر مع لغات التصريف الأخرى، تترجم عبارات الكوبول إلى وحدة تجميعية على مستوى الآلة.



ومن الخيارات المُتاحة استخدام مفسّر. فالمؤول أو المصرف يقرأ برنامجاً مصدرياً كاملاً ويولد وحدة تجميعية كاملة، ومن ناحية أخرى يعمل المفسّر على عبارة مصدرية واحدة كل مرة فيقرأها ويترجمها الى مستوى الآلة وينفذ التعليمات الثنائية الناتجة عنها، ومن ثم ينتقل الى العبارة المصدرية التالية. وتولّد المصرفات والمفسّرات تعليمات على مستوى الآلة، لكن طريقة المعالجة مختلفة.

ولكل لغة قواعدها الخاصة بتركيب الجمل واستعمال النقط والفواصل والتهجئة، فلا معنى على سبيل المثال، لبرنامج باسكال مصدري في مصرف كوبول أو مفسّر بيسيك. غير أنها جميعها تساعد في كتابة البرامج، ويظل هدف المبرمج هو نفسه أياً كانت اللغة التي تستخدم: تحديد سلسلة من الخطوات لتوجيه الكمبيوتر خلال عملية معيّنة.

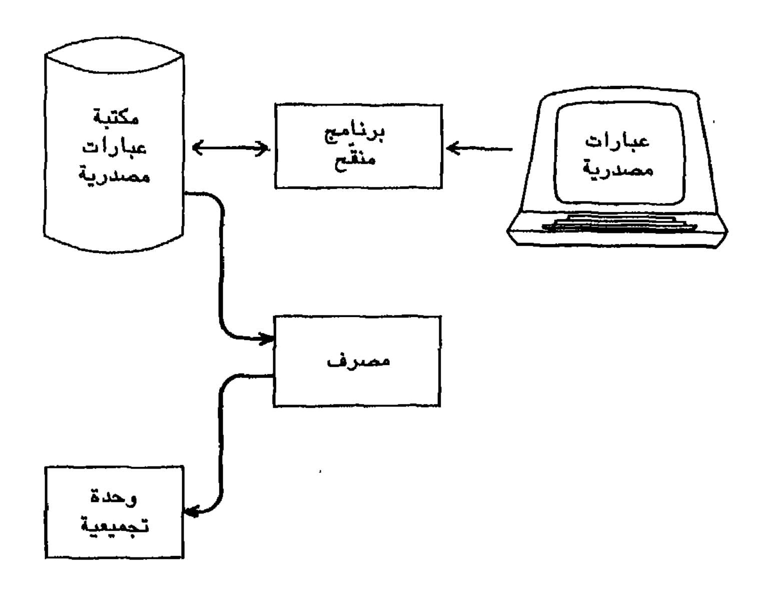
اللغات غير الاجرائية

يحدّد المبرمج بواسطة المؤولات والمصرفات والمفسّرات التقليدية الاجراء الذي يبلغ الكمبيوتر بدقة بكيفية حلّ مسألة معيّنة. ومع ذلك يحدّد المبرمج ببساطة بواسطة لغة من **اللغات غير الاجرائية** الحديثة (تُسمى بعض الأحيان بلغة الجيل الرابع أو اللغة الاعلانية)، البنية المنطقية للمسألة ويترك لمترجم اللغة تصور كيفية حلها. وتشمل نماذج اللغات غير الاجرائية المتوافرة تجارياً لغات برولوغ (Prolog) وفوكاس (Focus) ولوتس (Lotus) 1-2-3 وكثير من اللغات الأخرى، وهي تزداد رواجاً.

المكتبات

تصوّر مبرمجاً يكتب برنامجاً ضخماً. بعد طبع العبارات المصدرية يجري التحكّم بها بواسطة برنامج منقّح وتُخزّن على قرص. ويتوقف المبرمج في نهاية الأمر عن العمل ويزيل القرص من المدوار لأنه يندر أن تُكتب البرامج الضخمة في جلسة واحدة. ثم يُعاد إدخال القرص لاحقاً عندما يستأنف العمل، وتُضاف عبارات مصدرية جديدة الى العبارات القديمة. وقد يحتفظ ذلك القرص نفسه ببرامج مصدرية وحتى بأنهج أخرى كتبها مبرمجون اخرون، وهو، (أي القرص) مثال جيد على **مكتبة** عبارات مصدرية (شكل 7.6).

الشكل 7.6 يطبع المنقّح عادة العبارات المصدرية ويتحكّم بها، ويتم خزنها على مكتبة عبارات مصدرية. ويستطيع المبرمج قراءة العبارات الأصلية من المكتبة وتغييرها وحذفها وإضافة عبارات جديدة وتحديث المكتبة إذا كان هناك من تغييرات ضرورية. ولاحقاً تصرف العبارات المصدرية وتشكّل وحدة تجميعية.

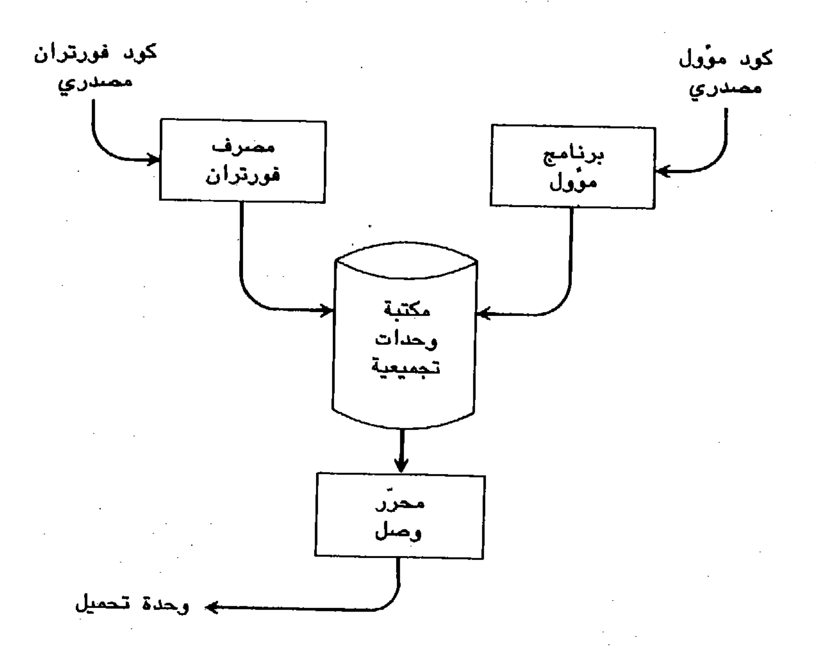


ولاحقاً ينتهي البرنامج ويصرف. وقد تحمل الوحدة التجميعية الناتجة مباشرة في الذاكرة الرئيسية، لكنها تخزّن في أغلب الأحيان على مكتبة وحدات تجميعية (شكل 7.7). وليس هناك من اختلاف جوهري بين وحدة تجميعية ينتجها مؤول وأخرى ينتجها مصرف فورتران (أو أي مصرف آخر يستخدم لهذا الشأن) لأن الوحدات التجميعية هي أنهج ثنائية على مستوى الآلة. وبالتالي يمكن خزن وحدات تجميعية مولّدة بلغات مصدرية مختلفة على المكتبة نفسها.

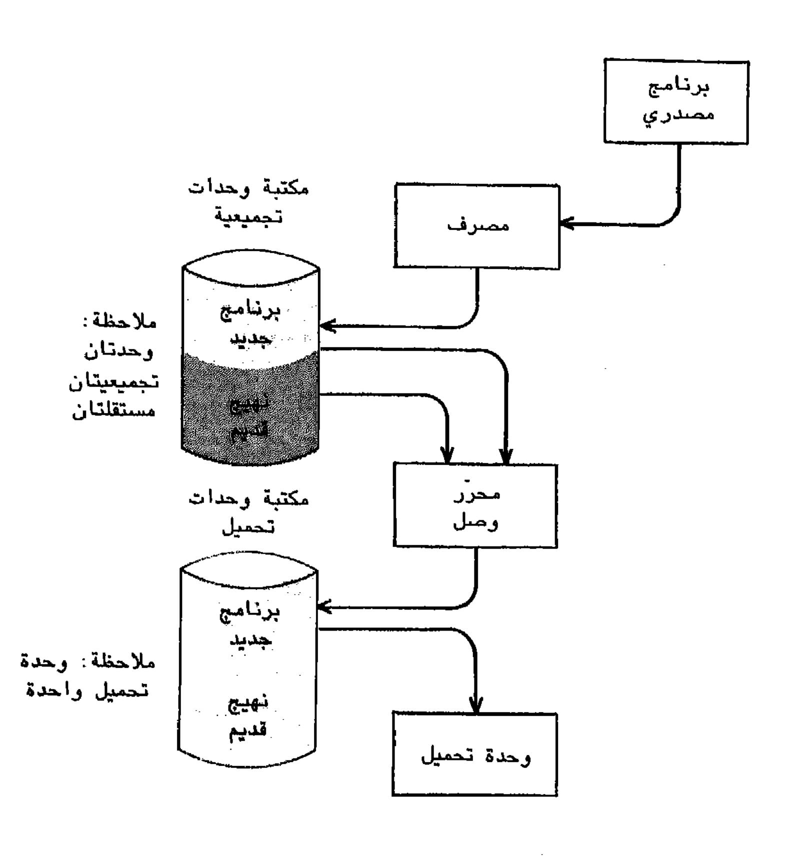
ويمكن تحميل بعض الوحدات التجميعية في الذاكرة وتنفيذها، غير أن بعضها يتضمّن إحالات إلى نهيجات لا تشكّل جزءاً من الوحدة التجميعية. تخيّل، على سبيل المثال، برنامجاً يسمح لكمبيوتر بمحاكاة لعبة ورق (شدة أو كوتشينة)، فسيكون استخدام المنطق نفسه مفيداً إذا كان مبرمج آخر قد كتب في وقت سابق نهيجاً متفوقاً لتوزيع الورق.

تخيّل البرنامج الجديد بعد كتابته وتصريفه وخزنه على مكتبة وحدات تجميعية (شكل 7.8). إن النهج الذي يوزّع الأوراق مخزون على المكتبة نفسها، وقبل تحميل البرنامج ينبغي جمع النهجين معاً ليشكّلا **وحدة تحميل**. إن الوحدة التجميعية هي ترجمة بلغة الآلة لوحدة مصدرية، وقد تتضمّن إحالات الى نهيجات أخرى، ووحدة التحميل هي برنامج كامل جاهز للتنفيذ تكون فيه جميع النهيجات في أماكنها الصحيحة. إن جمع الوحدات التجميعية لتشكيل وحدة تحميل هو مهمة الرابط التنفيذي أو المحمل (شكل 7.8).

الشكل 7.7 يمكن خزن الوحدات التجميعية على مكتبة أيضاً. وليس هناك من اختلاف جوهري بين وحدة تجميعية ينتجها مؤول وأخرى ينتجها مصرف فورتران لأن الوحدة التجميعية هي نهج ثنائي على مستوى الآلة، ولذلك يمكن خزن الاثنين معاً على المكتبة نفسها.



الشكل 7.8 يقرأ رابط تنفيذي أو محمل الوحدات التجميعية من مكتبة معيّنة ويربطها معاً ليشكّل وحدة تحميل.



وعموماً تشترى ألعاب الفيديو وبرامج صفحات القيد ومعالجات الكلمات وبرامج قواعد المعطيات ومناهج المحاسبة وغيرها من مجموعة البرامج التجارية على قرص في شكل وحدة تحميل. ونظراً لامكانية الاختيار بين الكود المصدري والوحدات التجميعية ووحدات التحميل، يجد أكثر الناس وحدة التحميل أسهل استخداماً لأن الكمبيوتر يستطيع بكل بساطة أن ينفذها بدون ترجمة. غير أنه من الصعب تغيير وحدات التحميل. فإذا كان المبرمج يخطط لتعديل أو تقليص تكلفة مجموعة برامجية، كان لا بدّ من الكود المصدري.

عملية تطوير البرامج

كيف يتعامل المبرمج مع تعديل مجموعة برامجية؟ وبصورة أعمّ، كيف يتعامل مبرمج محترف مع مهمّة كتابة برنامج أصلي؟ ليست البرمجة مجرد علم فقط، فهي تتمتع بلمسة فنية أيضاً، وبالتالي ليس من المفاجئ أن يعمل المبرمجون على اختلافهم بطرق مختلفة. ومع ذلك يبدأ معظمهم بعناية بتحديد المشكلة، ومن ثم يخطّطون حلولهم بالتفصيل قبل أن يكتبوا الكود. دعونا نحقّق في عملية تطوير البرامج.

تحديد المشكلة

إن الخطوة الأولى هي تحديد المشكلة. ويبدو ذلك أمراً بديهياً، لكن في أغلب الأحيان تكتب البرامج دون وجود فكرة واضحة عن دواعي الحاجة إليها. فحل مسألة خاطئة عقيم حتى ولو كان حلاً عظيماً.

تكتب البرامج لأن الناس بحاجة إلى معلومات، ولذلك يبدأ المبرمج بتعريف المعلومات المرغوبة، ومن ثم تحدد الخوارزميات أو القواعد لتوليد تلك المعلومات. وبعد أن نعطى المعلومات المرغوبة (الخرج) والخوارزميات يمكن عندئذ معرفة معطيات الدخل الضرورية. وتكون النتيجة تحديداً واضحاً للمشكلة يعطي المبرمج فكرة جيدةعما يجب أن ينجزه البرنامج.

وتُعرف البرامج أحياناً في نص النظام. وسنعرض لتحليل الأنظمة وتصميمها في الفصل 9.

التخطيط

تحدّد الخوارزميات ما يجب عمله، والمهمة التالية هي تقرير كيفية عمله. إن الهدف هو أن نذكر حلاً للمشكلة بلغة يمكن للكمبيوتر فهمها، ويستطيع الكمبيوتر أن يؤدّي وظائف الحساب والمقارنة والنسخ وطلب الدخل أو الخرج، وبالتالي يكون المبرمج محدداً بهذه العمليات الأساسية. إن إحدى نقاط البدء الجيدة هي حل نص صغير من المشكلة، ويستطيع المبرمج أن يكون فكرة جيدة عن الخطوات المطلوبة لبرمجة الخوارزميات من خلال حلّها فعلياً، ولو على نطاق محدود.

ويستعمل المبرمجون عدداً من الأدوات للمساعدة في تحويل حّل المشكلة إلى مصطلحات كمبيوترية. فمخطّطات سير العمليات، على سبيل المثال، يمكن أن تظهر منطق البرنامج بيانياً، ويستطيع المبرمج باستعمال كود زائف أن يعد «مسودة» المنطق قبل أن يحوّله الى كود مصدري. وغالباً ما يكتب مبرمجان أو أكثر البرامج الأكثر تعقيداً أو تلك التي تتضمّن قدراً كبيراً من المنطق، وعادة تجزأ هذه البرامج إلى وحدات أصغر أحادية الوظيفة يمكن تكويدها بشكل مستقل. ويخطّط المبرمج الجيد محتويات كل وحدة ويحدد العلاقات بين الوحدات قبل البدء بكتابة الكود المصدري. ويعد المبرمج خطة مفصّلة للبرنامج قبل البدء بكتابة الكود تماماً كما يهيىء المقاول رسوماً تنفيذية مفصّلة قبل البدء ببناء منزل.

كتابة البرنامج

أثناء التطبيق، يترجم المبرمج حل المشكلة إلى سلسلة من العبارات المصدرية المكتوبة بلغة برمجة معيّنة. وفي حين تنفرد كل لغة بقواعدها الخاصة بتركيب الجمل واستعمال النقط والفواصل والتهجئة - وكما أن تعلم لغة جديدة يستغرق وقتاً طويلاً - فإن كتابة التعليمات هي مهمّة آلية في الأساس. إن سرّ البرمجة الحقيقي ليس في تكويد التعليمات بكل بساطة، بل هو في معرفة أية تعليمة تكود لاحقاً، وهذا ما يتطلب منطقاً، ولحسن الحظ فإن معرفة كيفية البرمجة ليست شرطاً مسبقاً لاستخدام الكمبيوتر.

كشف الخطأ وتصحيحه والتوثيق

ما أن ينتهي تكويد البرنامج حتى يتوجّب على المبرمج أن يبدأ **بكشف الخطا فيه وتصحيحه**. إن الخطوة الأولى في أكثر الأحيان هي تصحيح أخطاء الأسلوب كالاستعمال الخاطئ للنقط والفواصل أو التهجئة، ويحدّد المصرف أو المفسر أماكن هذه الأخطاء، أمّا الأمر الأكثر صعوبة بكثير فهو العثور على الأخطاء المنطقية أو **الشوائب** التي تنجم عن تكويد تعليمة خاطئة وتصحيحها. ولا يكفي أن تكون التعليمات صحيحة، إذ يجب أن تكون التعليمات المناسبة في الترتيب المناسب. ومرة أخرى نقول أن التخطيط الدقيق هو السرّ لأن التخطيط الجيد يبسط عملية كشف الخطأ وتصحيحه.

ويتألف توثيق البرنامج من رسوم وملاحظات ومواد وصفية أخرى تفسّر الكود أو توضحه، وللتوثيق قيمة لا تثمّن في أثناء كشف خطأ البرنامج وتصحيحه كما أنه ضروري لتوفير صيانة برامج فعّالة، والأكثر إفادة هي الملاحظات التي تظهر في تدوين لوائح البرامج وتفسير المنطق.

الصيانة

ما أن يجهز البرنامج حتى تبدأ **الصيانة**. وبما أنه من المستحيل اختبار عدد كبير من البرامج الضخمة بشكل كامل، فإن بعض الشوائب قد تنفذ من مرحلة كشف الخطأ وتصحيحه، لتعود وتظهر بعد أشهر أو حتى سنوات لاحقاً. إن تحديد أماكن هذه الشوائب هو من مهام الصيانة الرئيسية. والأمر الأكثر أهمية هو الحاجة الى تحديث برنامج لإبقائه جارياً، فبرنامج كشف الرواتب، على سبيل المثال، ينبغي أن يحدث باستمرار لأن معدلات ضريبة الدخل تتغير تكراراً. إن سر الصيانة الناجح يكمن في التخطيط الدقيق والتوثيق وتصميم البرامج الجيد.

إعداد برامجك الخاصة

إن معرفة العزف على آلة موسيقية ليست ضرورية للاستمتاع بالموسيقى. وكذلك فإن معرفة كيفية البرمجة ليست شرطاً مسبقاً لاستخدام الكمبيوتر، فمعظم مستخدمي الكمبيوتر لا يستطيعون أن يبرمجوا.

يبقى أن معرفة كيفية البرمجة يمكن أن تجعلك مستخدماً أكثر كفاءة للكمبيوتر، تماماً كما تزيد المعرفة الأولية للآلة الموسيقية من تقديرك للموسيقى. وغني عن القول بالطبع أن معرفة البرمجة ضرورية إذا كنت تأمل أن تعتاش كمحترف كمبيوتر. ويجد بعض الناس البرمجة يسيرة بينما يجدها آخرون عسيرة للغاية. إن السر هو في الممارسة، فالطريقة الوحيدة لتعلّم كيفية البرمجة هي البرمجة.

لم يصمّم هذا الكتاب لتعليم البرمجة لكنه يضع الأسس المتينة لتعلّمها. إن التعرّف على البرمجة يجب أن يكون الخطوة التالية في دراستك.

الخلاصة

البرنامج هو سلسلة من التعليمات التي توجّه الكمبيوتر خلال عملية، وكل تعليمة تأمر الآلة بتأدية احدى وظائفها الأساسية. وينبغي أن يكون البرنامج المخزون في ذاكرة الكمبيوتر الرئيسية في شكل ثنائي لأن الكمبيوترات هي آلات ثنائية، غير أن عدد المبرمجين الذين يكتبون تعليمات على مستوى الآلة قليل جداً عملياً.

يكتب مبرمج لغة التأويل تعليمة واحدة مختصرة لكل تعليمة على مستوى الآلة ومن ثم يقرأ برنامج مؤول العبارات المصدرية ويترجمها إلى عبارات ثنائية ويولد وحدة تجميعية. ويقرأ المصرف العبارات المصدرية ويترجم كلا منها إلى تعليمة واحدة أو عدة تعليمات على مستوى الآلة ومن ثم يربطها معاً ليشكّل وحدة تجميعية. ويتعامل المفسّر مع عبارة مصدرية واحدة كل مرة/ فيترجمها وينفذ الكود الذي نتج عنها على مستوى الآلة قبل الانتقال إلى التعليمة التالية. ويحدّد المبرمج إجراء كاملاً لحل مشكلة بواسطة مصرف أو مفسّر أو مؤول تقليدي، ويستطيع المبرمج ببساطة باستعماله لغة من اللغات غير الاجرائية الحديثة أن يحدّد البنية المنطقية للمشكلة وأن يترك للبرنامج المترجم تصور كيفية حلّها.

ما أن تكتب البرامج حتى يخزّن الكود المصدري على مكتبة عبارات مصدرية، ويقرأ المؤول أو المصرف الكود المصدري ويخزّن الوحدة التجميعية التي نتجت على مكتبة وحدات تجميعية. ويجمع رابط تنفيذي أو محمل الوحدات التجميعية ليشكل وحدة تحميل. إن وحدة التحميل هي ترجمة على مستوى الآلة لكود المبرمج المصدري الذي يمكن أن يتضمّن إحالات إلى نهيجات أخرى، ووحدة التحميل هي برنامج كامل جاهز للتنفيذ.

تبدأ عملية كتابة برنامج أصلي بتحديد المشكلة. إن هدف المبرمج هو التعرّف إلى المعلومات المرغوبة وتحديد الخوارزميات المطلوبة لتوليد تلك المعلومات، ومن ثم تعريف معطيات الدخل التي تشغل الخوارزميات. بعد ذلك يأتي التخطيط، حيث تحدد الخطوات المنطقية الأساسية المطلوبة لحل المشكلة.

تتطلّب كتابة البرنامج ترجمة دقيقة للخطة إلى لغة برمجة، وتُسمّى عملية إزالة الأخطاء (أو الشوائب) من برنامج ما بكشف الخطأ وتصحيحه. وما أن يجهّز البرنامج حتى تبدأ الصيانة. ويصبح كشف أخطاء البرنامج وتصحيحها وصيانة البرنامج بسيطين مع التوثيق الجيّد.

ينبغي أن يكون تعلّم البرمجة هو المرحلة التالية من دراستك بعد إنهائك هذا الكتاب، والطريقة الوحيدة لتعلّم البرمجة هي التمرّس في البرمجة.

مصطلحات أساسية

- ☐ مؤول (مترجم جامع)
- ☐ شائبة
- ☐ مصرف
- ☐ كشف الخطأ وتصحيحه
- ☐ توثيق
- ☐ تعليمة
- ☐ مفسّر
- ☐ مكتبة
- ☐ وحدة تحميل
- ☐ لغة الآلة
- ☐ صيانة
- ☐ لغة غير إجرائية
- ☐ وحدة تجميعية
- ☐ تحديد المشكلة
- ☐ برنامج
- ☐ كود مصدري

اختبار ذاتي

1. في أثناء دورة آلة واحدة يستحضر المعالج وينفّذ ________.

أ. تعليمة واحدة
ب. عدة تعليمات
ج. عبارة واحدة
د. عدة عبارات

2. البرنامج الذي يتحكّم عملياً بالكمبيوتر يجب أن يكون مخزوناً في الذاكرة الرئيسية في شكل ________.

أ. ست عشري
ب. ثنائي
ج. منطقي
د. تسلسلي

3. عند استعمال ________، يكتب المبرمج تعليمة واحدة لكل تعليمة على مستوى الآلة.

أ. المؤول (المترجم الجامع)
ب. المفسّر
ج. المصرف
د. مولد البرامج

4. يكتب المبرمجون الكود ________.

أ. التجميعي
ب. المصدري
ج. على مستوى الآلة
د. لوحدة التحميل

5. تقرأ المصرفات والمؤولات كود المبرمج وتولد ________.

أ. وحدة تجميعية
ب. نهيجات
ج. وحدة تحميل
د. عبارات مصدرية

6. عند استخدام ________، تحوّل كل عبارة مصدرية إلى تعليمة أو أكثر على مستوى الآلة.

أ. مصرف
ب. كود آلي
ج. مؤول (مترجم جامع)
د. وحدة تجميعية

7. عند استخدام ________، تقرأ عبارة مصدرية واحدة وتترجم مباشرة إلى مستوى الآلة وتنفّذ من ثم قبل أن تقرأ العبارة المصدرية التالية.

أ. مصرف
ب. مفسّر
ج. مؤول (مترجم جامع)
د. لغة غير إجرائية

8. عند استخدام ________، يعرف المبرمج ببساطة البنية المنطقية لمسألة معيّنة.

أ. مصرف
ب. مفسّر
ج. مؤول (مترجم جامع)
د. لغة غير إجرائية

9. يخزّن كود المبرمج على ________.

أ. مكتبة مصدرية
ب. مكتبة تحميل
ج. مكتبة تجميعية
د. مكتبة مستخدم

10. يجمع الرابط التنفيذي بين وحدات ________ ليشكّل وحدة ________

أ. مصدرية/تجميعية
ب. مصدرية/تحميل
ج. تجميعية/تحميل
د. تجميعية/مصدرية

11. الخطوة الأولى في عملية تطوير البرامج هي ________.

أ. تحديد المشكلة
ب. التحليل
ج. التخطيط
د. التكويد

12. يبدأ تحديد المشكلة بـ ________ المرغوبة.

أ. المعلومات (الخرج)
ب. المعطيات (الدخل)
ج. المنطق
د. الخوارزميات

13. تُسمّى عملية أزالة الأخطاء من البرنامج بـ ________.

أ. التوثيق
ب. كشف الخطأ وتصحيحه
ج. التصريف
د. الصيانة

14. تُسمّى الرسوم البيانية والملاحظات وغيرها من المواد الوصفية التي تفسّر كود المبرمج بـ ________.

أ. الكود المصدري
ب. الكود التجميعي
ج. التوثيق
د. الخوارزمية

15. ما أن ينجز العمل في برنامج حتى تبدأ عملية ________.

أ. التوثيق
ب. كشف الخطأ وتصحيحه
ج. الصيانة
د. التكويد

الاجابات

1.أ 2.ب 3.أ 4.ب 5.أ 6.أ 7.ب 8.د 9.أ 10.ج 11.أ 12.أ 13.ب 14.ج 15.ج.

ربط المفاهيم

1. الكمبيوتر بدون برنامج يؤمّن التحكم ليس أكثر من آلة حاسبة باهظة التكاليف. هل توافق على ذلك؟ لماذا توافق ولماذا لا توافق؟
2. أربط فكرة التعليمة بدورة الآلة الأساسية في الكمبيوتر.

3. لماذا تعتبر لغات البرمجة ضرورية؟
4. فرّق بين المؤول (المترجم الجامع) والمصرف.
5. فرّق بين المصرف والمفسّر.
6. ما هي المكتبة؟ ولماذا تعد المكتبات مفيدة؟
7. فرّق بين الوحدة المصدرية والوحدة التجميعية ووحدة التحميل.
8. عدّد الخطوات في عملية تطوير البرامج واشرح باختصار ما يحدث خلال كل خطوة.
9. يتطلّب تطوير البرنامج عملية منهجية خطوة بعد خطوة. لماذا ؟
10. الطريقة الوحيدة لتعلّم البرمجة هي البرمجة. لماذا ؟

8.

إدارة المعطيات

مفاهيم أساسية

لماذا إدارة المعطيات؟

نيل المعطيات

- ☐ بنى المعطيات
- ☐ تحديد مواقع الملفات
- ☐ تحديد مواقع السجلات
- ☐ مفهوم السجل النسبي

أساليب النيل

إدارة قاعدة المعطيات

لماذا إدارة المعطيات؟

بحثنا في الفصلين السابقين الكيان المادي والكيان المنطقي بشيء من التفصيل. وفي هذا الفصل، نوجه انتباهنا الى مورد الكمبيوتر الأساسي الثالث، **المعطيات**. ويتطلب الكثير من تطبيقات الكمبيوتر خزن المعطيات حتى تعالج في وقت لاحق. على ان مجرد خزن المعطيات ليس كافياً. إن نظام الكمبيوتر النموذجي، حتى وإن كان صغيراً، يمكن أن يكون له عشرات الأقراص والأشرطة، التي يخزن كل منها معطيات لعشرات الاستعمالات المختلفة. وللقيام بأي تطبيق معين، فإن مجموعة واحدة، وواحدة فقط، من المعطيات ستكون كافية. ويجب أن نكون قادرين على خزن المعطيات المعينة التي يحتاجها برنامج معين، وتحديد موقعها، واسترجاعها، وذلك هو ما تعنى به **إدارة المعطيات**.

نيل المعطيات

تخيّل قريصاً واحداً يحتوي على عدد من البرامج. فبالنسبة الى تطبيق معين، واحد من هذه البرامج يكفي. ولكن كيف يتم اختيار برنامج معين، ثم يحمل وينفذ؟ تعلمنا في الفصل السادس أن برنامج التشغيل، مستجيباً لأمر المستخدم، يقرأ فهرس القرص، ويبحث فيه عن اسم البرنامج المطلوب، ويستخرج مسار البرنامج وعنوان قطاعه، ومن ثم يصدر اوامر بدائية ليقرأه في الذاكرة الرئيسية. وفيما بعد، وبعد أمر RUN، يعطى البرنامج تحكماً من المعالج.

إن نيل المعطيات يشكّل مشكلة مشابهة. إذ يمكن أن يحمل قريص واحد معطيات لاستعمالات عديدة مختلفة. وبالنسبة الى تطبيق معين، فإن مجموعة واحدة، وواحدة فقط، من المعطيات ستكون كافية، والعثور على المعطيات الصحيحة يشبه إلى حد كبير العثور على البرنامج الصحيح. وعلى كل حال هناك اختلاف بين نيل البرامج ونيل المعطيات. فعندما تنشأ الحاجة الى برنامج، يجب أن تحمل كل تعليماته في الذاكرة. ومن الناحية الأخرى، فإن المعطيات تعالج نموذجياً بالانتقاء، عناصر قليلة في المرة الواحدة. وهكذا، فإن مجرد تحديد مواضع المعطيات لا يكفي في حد ذاته: يجب أن نكون قادرين على التمييز بين **عناصر المعطيات** المنفردة أيضاً.

بنى المعطيات

إن مفتاح استرجاع المعطيات هو تذكّر الموضع الذي تخزّن فيه. فإذا خزّنت عناصر المعطيات وفقاً لبنية متماسكة ومفهومة جيداً، أمكن استرجاعها بتذكر تلك البنية.

واللائحة هي أبسط **بنية معطيات**. على سبيل المثال، يمكن ان تخزّن معطيات برنامج يعد متوسطاً في شكل سلسلة من الأرقام مفصولة بشاولات (الشكل 8.1). وتميّز الشاولات عناصر المعطيات الفردية.

وتتحمّل معظم لغات البرمجة بنى معطيات أكثر تعقيداً تُسمى **صفيفات** (الشكل 8.2). ويمكن أن يتضمن كل عنصر صفيف قيمة معطيات واحدة. ويعين لكل عنصر رقم معرف منفرد أو عدة أرقام، ويمكن إدخال عنصر المعطيات المفرد، او استخراجه أو معالجته بالاحالة إلى تلك الأرقام. على سبيل المثال، يتم التعرّف على العناصر في الصفيف الموضح في الشكل 8.2 برقم الصف ورقم العمود ويحتوي الصف 1 العمود 3 (العنصر 1,3) القيمة 29. وفي اللحظة التي يملأ فيها صفيف، يمكن أن يكتب إلى قرص أو شريط أو أي وسط ثانوي ويقرأ ثانية، فيما بعد، الى الذاكرة للمعالجة.

لنأخذ مثلاً برنامجاً يولد بطاقات عناوين وأسماء. فنحتاج لكل بطاقة اسماً وعنوان شارع ومدينة وولاية ورقم المنطقة. وإذا كنا بحاجة الى بطاقات قليلة فحسب، فقد نخزّن المعطيات في لائحة، ولكن سرعان ما يصبح فصل العناصر أمراً مضجراً. والخيار هو إنشاء صفيف من الأسماء والعناوين ويحمل كل صف المعطيات الخاصة ببطاقة واحدة. أمّا المشكلة الوحيدة فهي أن الصفيف بأسره يجب أن يكون في الذاكرة الرئيسية قبل أن يصبح من الممكن نيل العناصر الفردية، وحيّز الذاكرة الرئيسية محدود. وهكذا، وحتى باستعمال الصفيف فاننا لا نستطيع أن نولد إلاّ بطاقات قليلة نسبياً.

والحل الأفضل هو تنظيم المعطيات في شكل ملف (الشكل 8.3). تبدأ كل معطيات الكمبيوتر كنمط من الخوينات. وفي الملف تجمع الخوينات لتشكل رموزاً. وتشكل مجموعات الرموز بدورها عناصر معطيات ذات معنى تُسمى «**حقولاً**». وتُسمى مجموعة الحقول ذات العلاقة «**سجلا**». **والملف** هو مجموعة من السجلات مترابطة. فمثلاً نجد ان في ملف أسماء وعناوين، يكون اسم الفرد حقلاً. ويتضمن كل سجل مجموعة كاملة من المعطيات الخاصة بفرد واحد (الاسم، العنوان، الشارع، وهكذا). كما ان الملف يتألف من سجلات.

الشكل 8.1 اللائحة هي أبسط بنى المعطيات. وتعمل الفواصل، مثل هذه الشاولات، على التمييز بين القيم الفردية. وغالباً ما تحدّد قيمة «المعلم» مثل الرقم السلبي نهاية هذه اللائحة.

4410, 843, 184, 31, 905, 6357 44, 7702, 228, 59, −1

الشكل 8.2 تساند معظم لغات البرمجة بنى معطيات اكثر تعقيداً تُسمّى صفيفا وتخصص خلايا فردية لعدد او أعداد، ثم تدخل قيم المعطيات وتعالج وتستخرج بالرجوع الى هذه الأعداد.

1,1 71	1,2 38	1,3 29	1,4 90	1,5 70
2,1 91	2,2 13	2,3 56	2,4 77	2,5 20
3,1 68	3,2 18	3,3 54	3,4 63	3,5 56
4,1 12	4,2 38	4,3 68	4,4 39	4,5 74
5,1 82	5,2 80	5,3 35	5,4 98	5,5 61

الشكل 8.3 تجمع الرموز لتشكل حقولاً. وتجمع الحقول لتشكل سجلات. والملف هو مجموعة من السجلات ذات العلاقة.

	الاسم	العنوان	المدينة	الولاية	رقم المنطقة
ملف	م. آتكنز	142 مييل	أكسفورد	أوهايو	450781718
	ش. بيكر	713 مبين	سنسناتي	أوهايو	457033304
	ت. بيتس	42 ساوث بولوفارد	اتلانتا	جورجيا	352170315
	ل. كامبانلا	8 تاور سكوير	سان هوزيه	كاليفورنيا	953214450
	ش. شان	ستيت راوت 77	بنغماتون	نيويورك	127561495
	ت. غارسيا	473 ديكسي هايوي	ليكسينغتون	كينتاكي	434101236
	⋮	⋮	⋮	⋮	⋮
	آ. وايت	نورثسايد مول	اورلاندو	فلوريدا	214504372
	سمة				
	حقل	حقل	حقل	حقل	حقل
	سجل				

تعالج المعطيات في الملف سجلاً بسجل. وعادة، يخزّن الملف في وسط ثانوي كالقرص. وتكتب البرامج لتقرأ سجلاً، ولتعالج ملفاته وتولد الخرج الملائم ثم تقرأ وتعالج سجلاً آخر. ولأن سجلاً واحداً فقط يتواجد في الذاكرة الرئيسية في كل مرة، فاننا نحتاج الى قدر قليل جداً من الذاكرة. وبما ان خزن عدد كبير من السجلات في القرص الواحد ممكن، نستطيع معالجة قدر كبير من المعطيات في هذه المساحة المحدودة.

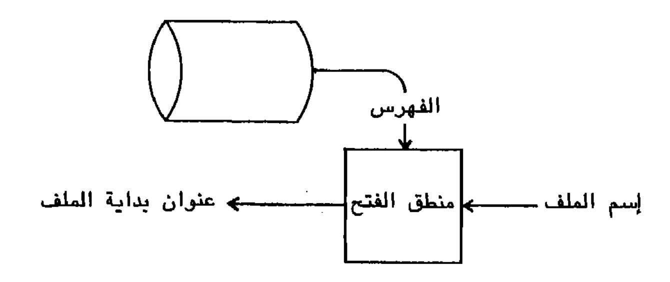
تحديد مواقع الملفات

إذا تخيّلنا ملفاً مخزوناً في قرص. فسنجد ان الخطوة الأولى لنيل معطياته تبدأ بايجاد الملف نفسه. وتشبه هذه المهمة الى حد كبير مهمة ايجاد برنامج، ولكن هنالك اختلافات. وبإتباع أوامر مثل LOAD تحميل وRUN تنفيذ برنامج، يحمل نظام التشغيل البرامج، ومن الناحية الأخرى تعالج المعطيات بتطبيق برامج في سياق منطق برنامج. ونموذجياً، يطلب البرنامج من نظام التشغيل فتح الملف قبل أن تطلب المعطيات مباشرة. فلكل ملف اسم؛ يقرأ منطق الفتح (الشكل 8.4) فهرس القرص، ويبحث فيه عن الاسم ويجد عنوان أول سجل في الملف.

تحديد مواقع السجلات

ما أن يتم تحديد موقع الملف، يمكن أن تبدأ عملية نيل سجلاته. وعندما يحتاج البرنامج إلى دخل معطيات، يقرأ سجلاً، كما انه يكون مستعداً لنتائج خرج، يكتب سجلاً. ويجدر بنا أن نلاحظ ان هذه التعليمات تتعامل مع سجلات مختارة، وليس مع الملف بأسره. نحن نفتح الملفات. ثم نقرأ ونكتب السجلات.

الشكل 8.4 عندما يفتح ملف ما، يقرأ فهرس القرص في الذاكرة الرئيسية ويبحث فيها عن اسم الملف المطلوب. وإذا وجد الاسم، يستخرج عنوان بداية الملف من الفهرس.



الشكل 8.5 يفكر المبرمجون على أساس مبدأ الدخل/الخرج المنطقي. ويستجيب جهاز مادي الى أوامر مادية بدائية. وبطريقة ما، يجب أن تترجم طلبات الدخل/الخرج المنطقية الى أوامر مادية.

برنامج تطبيقي
طلبات المبرمج دخل/خرج منطقية
?
أوامر بدائية مادية

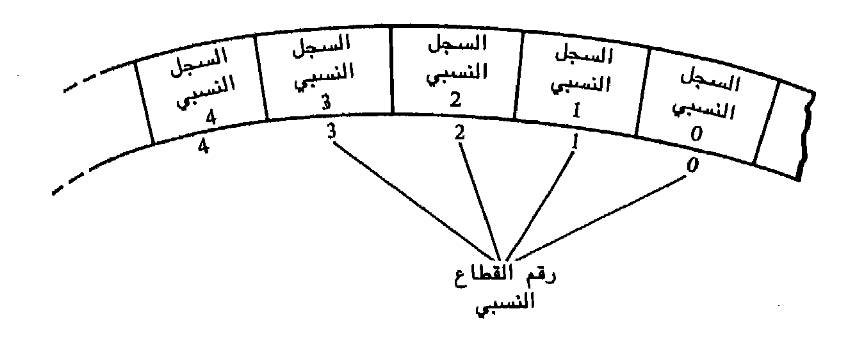
إذا تفحّصنا عملية نيل المعطيات. فنرى ان المبرمج ينظر الى المعطيات منطقياً ويتم طلب السجل التالي أو اسم وعنوان عميل معين. وتخزن المعطيات في وسيط ثانوي مثل القرص. ولكي يتمّ نيل السجل عملياً، يجب أن يعطى مدوار القرص مجموعة من الأوامر البدائية يقوم بما يمليه عليه من بحث وكتابة وقراءة. يفكر المبرمج على أساس مبدأ **الدخل/الخرج المنطقي**. ويخزن الجهاز الخارجي القطاعات المادية ويسترجعها. أي أنه يفكر على أساس مبدأ **الدخل/الخرج المادي**. ويجب أن تكون هناك آلية لترجمة طلبات المبرمج المنطقية الى الأوامر المادية الملائمة. (الشكل 8.5). ففي الكمبيوترات الصغيرة، يوجد الكثير من المنطق في نظام التحكم الخاص بدخل/خرج نظام التشغيل. أمّا في الآلات الأكبر، فتستخدم وسائل النيل. وتترجم، بشكل متزايد، طلبات المبرمج المنطقية للمعطيات إلى شكل مادي بواسطة إدارة قاعدة المعطيات.

مفهوم السجل النسبي

كيف يمكن للكيان المنطقي سواء أكان نظام تشغيل أو أسلوب نيل أو كيان منطقي قائم على قاعدة المعطيات، أن يجد السجلات المعينة في ملف ما ؟ إن المفتاح لكثير من تقنية الخزن والاسترجاع هو **رقم السجل النسبي**. تخيل سلسلة من 100 سجل. رقم الأول بـ0 والثاني بـ1 والثالث بـ2 وهكذا. تشير الأعداد الى موقع سجل معين بالنسبة للسجل الأول في الملف. ان السجل الأول للملف (السجل النسبي 0) هو في «بداية الملف زائداً 0» وسجل الملف الثاني هو «في بداية الملف زائداً 1» وهكذا دواليك.

والآن أخزن السجلات على القرص (الشكل 8.6)، وللابقاء على المثال المبدئي الذي ضربناه بسيطاً، سنخزّن سجلاً واحداً مقابل كل قطاع. ثم رقم القطاعات بالنسبة لبداية الملف - 0 ,1 ,2 وهلم جرا. لاحظ ان رقم السجل النسبي، وهو مفهوم منطقي، ورقم القطاع النسبي، وهو موقع مادي، متطابقان. وإذا أعطينا رقم سجل نسبي، يكون ممكناً عد رقم قطاع نسبي. وإذا أعطينا رقم قطاع نسبي يكون ممكناً عد عنوان مادي على قرص.

الشكل 8.6 يشير رقم السجل النسبي الى موقعه بالنسبة الى السجل الأول في الملف. وتولد أرقام القطاعات النسبية بالعد من قطاع الملف المادي الأول. وإذا اعطينا رقم سجل نسبي يصبح ممكناً عد عنوان القرص المادي.



الشكل 8.7 يمكن حساب عنوان قرص مادي إذا اعطينا بداية عنوان ملف (من افتتح) ورقم سجل نسبي.

رقم السجل النسبي	الموقع الفعلي على القرص: السكة	القطاع
0	30	0
1	30	1
٢	30	2
3	30	3
4	30	4
5	30	5
6	30	6
7	30	7
8	30	8
9	30	9
10	30	10
⋮	⋮	⋮

افترض ان ملفاً يبدأ عند المسار 30، القطاع 0، وان سجلاً منطقياً واحداً مخزون في كل قطاع. وكما يوضح الشكل 8.7، فإن السجل النسبي 0 مخزون في المسار 30، القطاع 0، والسجل النسبي 1 في المسار 30، القطاع 1 وهلم جرا. أين يوجد السجل 10؟ يجب أن يكون مخزوناً في المسار 30، القطاع 10. وفي المثال الذي ضربناه، فإن رقم السجل النسبي يشير الى بعد موقع خزن السجل عن بداية الملف. وهكذا، فإننا نستطيع أن نحسب الموقع المادي لأي سجل بإضافة رقم سجله النسبي الى بداية عنوان الملف (التي، ان كنت تذكر، قد استخرجت من فهرس القرص عندما فتح الملف). يبدأ الملف عند المسار 30، القطاع 0. وقد خزن السجل النسبي 10 على بعد 10 قطاعات، عند المسار 30، القطاع 10. لقراءة السجل 10، يجب أن يخبر مدوار القرص أن يبحث عن المسار 30 ويقرأ القطاع 10. لقد ترجمنا طلب معطيات منطقية الى أوامر مادية محددة.

قد تتعقّد المسائل إذا نحن خزّنا سجلين منطقيين أو أكثر في كل قطاع، أو اذا أعددنا ملفاً يمتد على مسارين أو أكثر. وفي حين اننا لن نبحث التفاصيل، فإنه ممكن في أي من الحالين تطوير خوارزمية بسيطة لحسب موقع سجل مادي، إذا أعطينا رقمه النسبي. هناك خوارزميات متنوّعة يمكن استعمالها. ويسمح بعضها للسجلات بأن تخزّن أو تسترجع بالتعاقب. والبعض الآخر يسمح للسجلات الفردية بأن تنال بترتيب عشوائي. إذن دعنا ندرس عدداً قليلاً من الأساليب الفنية الشائعة لنيل المعطيات.

أساليب النيل

تخيّل انك تعد بلاغات اجتماع لناد. وانك بحاجة الى مجموعة من بطاقات الارسال بالبريد، وإسم كل عضو وعنوانه مسجل على بطاقة فهرس. لربما كانت أسهل طريقة لتوليد البطاقة هي نسخ المعطيات من البطاقة الأولى والرجوع الى البطاقة الثانية ونسخ ما بها من معطيات، وهكذا، تعالج البطاقات بالتتالي من بداية الملف الى نهايته.

يواجه ناشرو المجلات المشكلة نفسها مع إصدار كل عدد جديد، إذ انهم يحتاجون الى بطاقات ارسال بالبريد لعشرات الألوف من المشتركين. وبدلاً من استعمال بطاقات الفهرس، يخزّنون معطيات العميل على قرص أو شريط مغنطيسي، لكل مشتر كسجل واحد. أسهل طريقة للتأكد من أن جميع البطاقات ستولد هي في معالجة السجلات بالترتيب الذي خزّنت به، والمباشرة بشكل متعاقب من السجل الأول في الملف حتى الاخير. ولكي نسهل القيام بالعملية، يمكن أن تصنف السجلات مسبقاً بكود بريدي أو منطقة ارسال بالبريد، ولكن الفكرة الأساسية لمعالجة المعطيات بالترتيب المادي ما تزال سليمة.

كيف يرتبط هذا بمفهوم رقم السجل النسبي؟ ان رقم السجل النسبي يشير الى موقع السجل في الملف. وبالنسبة الى **النيل بالتتالي**، تبدأ المعالجة بالسجل النسبي 0، ثم تتحرك الى السجل النسبي 1، 2، وهلم جرا. كما ان نيل المعطيات بالتتالي يشتمل على شيء أكثر بقليل من مجرد العدّ. على سبيل المثال، إذا انتهى برنامج لتوه من معالجة السجل النسبي 14، فما هو السجل التالي؟ من الواضح انه السجل النسبي 15. وقد سبق ورأينا بالفعل من قبل كيف ان من الممكن تحويل رقم سجل نسبي الى عنوان مادي؛ انه لمن الممكن، بمجرد عد السجلات، أن تقرأها أو تكتبها بالترتيب المادي.

ان معالجة السجلات بالتتالي ليست مقبولة دائماً. مثلاً، عندما ينتقل مشترك أو مشتركة الى منزل جديد يجب أن يترك عنوانه أو عنوانها في الملف. ان عملية البحث عن سجل ذلك المشترك هي بالتالي كالبحث عن رقم الهاتف بالصفحة الأولى من دليل الهاتف. إن هذه ليست الطريقة التي تستعمل بها دليل التلفونات. وبدلاً عن ذلك، فإننا على معرفة بأن السجلات مخزونة حسب الترتيب الأبجدي، وبالتالي نسرع بتضييق مساحة بحثنا

الى جزء من صفحة واحدة، ثم نبدأ بقراءة الأسماء والعناوين المدوّنة، متجاهلين معظم المعطيات. أمّا الطريقة التي تستخدم بها دليل التلفونات فهي خير مثال **للنيل العشوائي أو المباشر**.

يقرأ مدوار القرص او يكتب سجلاً واحداً في كل مرة. ولكي ينال المبرمج سجلاً معيناً عشوائياً، فإن كل ما عليه أن يفعله هو أن يتذكر عنوانه، وأن يطلبه. والمشكلة تكمن في تذكر عناوين كل هذه الأقراص. أمّا احد الحلول فهو الاحتفاظ **بفهرس** للسجلات. ومرة أخرى، سنستخدم إسم وعنوان الملف كمثال، عندما نرغب في نيل سجل فرد عميل بعينه. وفي الوقت الذي يجري فيه فتح الملف الجديد، تكتب السجلات، الواحد تلو الآخر، وبترتيب أرقام سجلات نسبية. وبالاضافة الى ذلك، وفي الوقت الذي يكتب فيه كل سجل، فإن اسم العميل ورقم السجل النسبي المرافق له يسجلان في صفيف أو فهرس (الشكل 8.8). وبعد كتابة آخر سجل الى القرص وتسجيل موقعه في الفهرس، يخزّن الفهرس نفسه.

في الوقت الذي يفتح الفهرس الجديد، يمكن أن يستعمل لايجاد السجلات الفردية. افترض، على سبيل المثال، ان سوزان سميث قد بدّلت عنوانها. ولكي يسجل عنوانها الجديد في الملف، يستطيع البرنامج أن:

1. يقرأ فهرس الملف،
2. يبحث في الفهرس عن اسمها،
3. يجد رقم سجلها النسبي،
4. يحسب عنوان القرص، ويقرأ سجلها،
5. يغير عنوانها،
6. يعيد كتابة السجل في نفس المكان في القرص.

لاحظ ان هذا السجل بعينه ينال مباشرة، وليس لأية سجلات أخرى في الملف علاقة به.

الشكل 8.8 يمكن لفهرس الملف أن يساعد عندما يكون من الضروري نيل السجلات مباشرة.

المفتاح	السجل النسبي
م. آتكنز	0
ش. بيكر	1
ت. بيتس	2
ل. كامبانلا	3
ش. شان	4
ت. غارسيا	5
.	.
.	.
.	.

الشكل 8.9 تترجم أساليب النيل طلبات المبرمج لدخل/خرج المنطقية الى أوامر مادية.

الفكرة الأساسية للنيل المباشر هي أن يخصّص لكل سجل مفتاح منطقي يسهل تذكّره، ثم تحويل ذلك المفتاح الى رقم سجل نسبي. وإذا أعطينا هذا الموقع النسبي، يمكن للعنوان المادي أن يحسب والسجل أن ينال. كما ان استخدام الفهرس هو واحد من الأساليب التقنية لتحويل المفاتيح الى عناوين مادية. والخيار هو تمرير المفتاح العددي الى خوارزمية وحساب رقم السجل النسبي. لكلا الأسلوبين التقنيين نفس الهدف: تحويل طلبات معطيات المبرمج المنطقية الى اتخاذ شكل مادي.

لقد تعرّفنا في وقت سابق في هذا الفصل (الشكل 8.5) على الفجوة التي تفصل بين دخل/خرج المنطقي والمادي. إن **أسلوب النيل** هو وحدة كيان منطقي يساعد على تخطي هذا التفاوت (الشكل 8.9)، محوّلة المفاتيح المنطقية الى عناوين مادية، ومصدرة الأوامر البدائية الملائمة. كما ان هنالك أنواعاً مختلفة كثيرة من تنظيمات متتالية ومفهرسة ومباشرة ولكل واحد قوانين نيل خاصة به. ان استخدام تنويعة من تقنية نيل المعطيات يمكن أن يؤدّي الى الارتباك، وهذا واحد من أسباب ازدياد شعبية نظام إدارة قاعدة المعطيات.

إدارة قاعدة المعطيات

هناك مشاكل بالنسبة لادارة المعطيات التقليدية. وينجم الكثير من هذه المشكلات عن النظر الى التطبيقات بشكل مستقل. خذ على سبيل المثال جدول الرواتب. فاننا نجد إن معظم المنظمات تعد جداول الرواتب بالكمبيوتر لأن استخدام الآلة بدلاً من جيش من الموظفين يحقّق اقتصاداً في النفقات. وهكذا، تضع الشركة برنامج جدول رواتب لمعالجة ملف جدول الرواتب. ولأن قائمة الموجودات والحسابات المستلمة والحسابات المدفوعة والتحليل العام لدفتر الأستاذ هي كلها تطبيقات متشابهة، فإن الشركة تضع برنامج قائمة وملف موجودات، وبرنامج حسابات مستلمة أيضاً وملف حسابات مستلمة وهلم جرا. وكل برنامج من هذه البرامج مستقل ويعالج كل منها ملف معطياته الخاص به.

لماذا يشكّل هذا مشكلة؟ من أحد الأسباب ان التطبيقات المختلفة تحتاج في الغالب الى نفس عناصر المعطيات. على سبيل المثال، تولد المدارس الفواتير وتقارير نتائج امتحانات الطلاب على حد سواء. أنظر الى كل تطبيق بشكل مستقل عن الآخر. فنجد ان برنامج إعداد الفواتير يقرأ ملفاً لمعطيات الفواتير، ويقرأ برنامج اعداد تقرير نتائج الامتحانات ملفاً مستقلاً به معطيات الدرجات. ويرسل خرج البرنامجين بالبريد الى منزل كل طالب، وهكذا يجب أن تسجل عناوين الطلاب بطريقة تزيد عن الحاجة في الملفين كليهما. وماذا يحدث عندما يغير طالب مكان إقامته؟ ما لم يحدث الملفان فإن أحدهما سيكون على خطأ. إن من الصعب حفظ **المعطيات المطنبة**.

والمشكلة الأخرى الأكثر حدة هي مشكلة **تبعية المعطيات**. فلكل أسلوب نيل قوانينه الخاصة لخزن واسترجاع المعطيات، وحيل خاصة «بالمهنة» يمكن أن تحسن بشكل

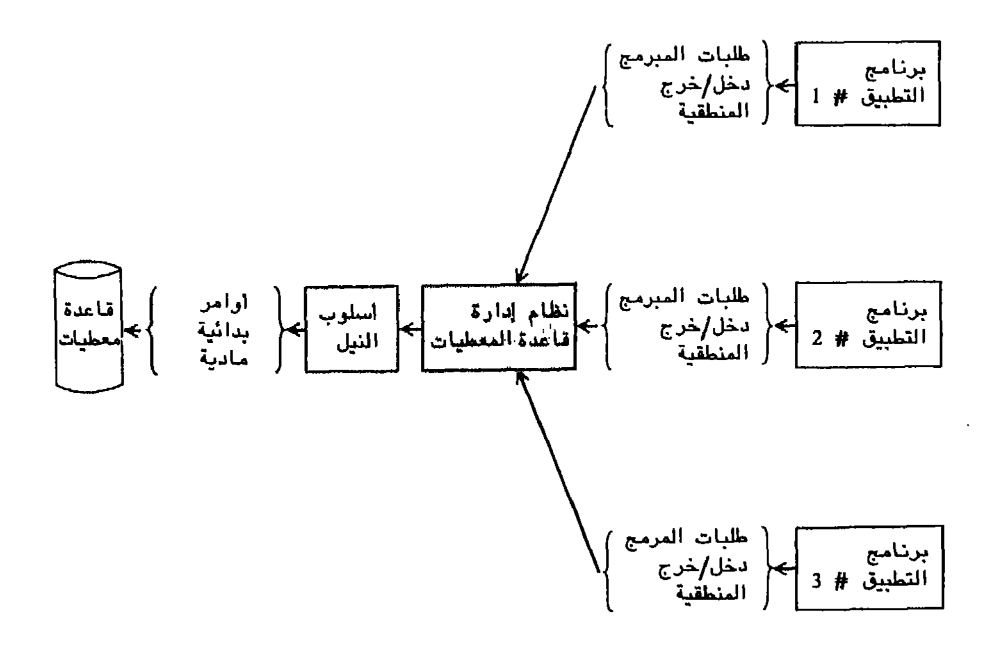
ملموس فعالية برنامج من البرامج. وبما ان الدافع لاستعمال الكمبيوتر هو الاقتصاد في النفقات، فإن المبرمج يتعرض دائماً لاغراء تحقيق المزيد من الاقتصاد في النفقات باستغلال هذه الفعاليات. وهكذا فإن منطق البرنامج يصبح أكثر اعتماداً على البنية المادية للمعطيات. وعندما يربط البرنامج ببنية معطياته المادية، فإن تغيير تلك البنية يكاد يكون من المؤكد انه يستوجب تغيير البرنامج. وكنتيجة لذلك، فإن من الصعب الاحتفاظ ببرامج تستعمل أساليب النيل التقليدية.

إن حل المشكلتين، يكون في الغالب، في تنظيم المعطيات **كقاعدة معطيات** أحادية متكاملة. ومن ثم يمكن تركيز مهمة التحكم في نيل كل المعطيات في **نظام ادارة قاعدة معطيات** مركزي (الشكل 8.10).

كيف يوفر استعمال قاعدة المعطيات المتمركزة الحل لمشكلة تكرار المعطيات؟ تجمع كل المعطيات وتخزّن في مكان واحد، وتبعاً لذلك توجد نسخة واحدة وواحدة فقط من كل عنصر معطيات معين. وعندما تتغير قيمة أي عنصر (عنوان مثلاً)، فإن قاعدة المعطيات الأحادية تصحّح. ويحصل أي برنامج يطلب نيل عنصر المعطيات هذا، على نفس القيمة، لأنه لا توجد إلاّ قيمة واحدة.

كيف تستطيع قاعدة المعطيات أن تحل مشكلة الاعتماد على المعطيات؟ بما ان مسؤولية نيل المعطيات المادية تقع على عاتق نظام قاعدة المعطيات، فإن المبرمج يستطيع أن يتجاهل البنية المادية للمعطيات. وكنتيجة لذلك، تميل البرامج لأن تكون أقل اعتماداً على معطياتها ويكون من الأسهل كثيراً بشكل عام الاحتفاظ بها. توقع ان يستمر اتجاه الأخذ بنظام ادارة قاعدة المعطيات.

الشكل 8.10 يعزل نظام إدارة قاعدة المعطيات المبرمج عن المعطيات المادية.



الخلاصة

كانت المعطيات هي محور هذا الفصل. إن خزن المعطيات ليس كافياً في حد ذاته، يجب أن تخزّن بطريقة يجب معها استرجاع عناصر المعطيات الفردية عندما تكون هنالك حاجة لها. ومفتاح استرجاع المعطيات هو تذكّر مكان خزنها. والسر هو، غالباً، خزنها في بنية محدّدة المعالم بشكل جيد ثم استرجاعها بتذكر تلك البنية. واللائحة هي أبسط بنى المعطيات. وتحتوي معظم لغات البرمجة بنى معطيات أكثر تعقيداً تُسمّى الصفيف، تعرف فيه عناصر المعطيات المفردة بترقيم خلايا ذلك الصفيف.

تجمع الرموز الفردية في الملف لتشكل حقولاً وتجمع الحقول لتشكل سجلات وتشكل مجموعة ذات علاقة من السجلات ملفاً. ويحدد موقع الملف بأن يطلب من نظام التشغيل أن يفتحه. وعندما ينفذ منطق واحد يجري البحث في فهرس القرص عن اسم الملف ويستخرج مسار وعنوان قطاع السجل الأول.

إن نيل المعطيات في ملف ما، يتطلّب قراءة وكتابة سجلات فردية. يفكر المبرمج بلغة دخل/خرج منطقية، ينجز الكيان المادي دخل/خرج مادي. الكيان المنطقي، في شكل نظام تحكم دخل/خرج الخاص بنظام تشغيل، وأسلوب نيل أو نظام إدارة قاعدة معطيات، يترجم الطلبات المنطقية الى أوامر مادية ضرورية.

غالباً ما يكون مفتاح ايجاد سجل معين هو رقم السجل النسبي. ترقم السجلات في الملف بالتتابع، ويشير رقم كل سجل نسبي الى موقع السجل النسبي من السجل الأول في الملف. وإذا عرفنا رقم السجل النسبي يكون من الممكن أن نحسب موقعه المادي.

يستعمل العديد من الأساليب الفنية لنيل المعطيات لتحويل الأعداد النسبية للسجل الى مواقع سجل مادية. وبالنيل المتتابع تخزّن المعطيات وتستعاد بنظام محدّد، أساساً بعد السجلات. كما يمكن استعادة السجلات الفردية سواء بأسلوب النيل المباشر أو العشوائي بدون اعتبار لمواقعها في الملف المادي. وغالباً يحفظ فهرس للسجلات. وعندما تنشأ الحاجة لسجل، يجري البحث في الفهرس ويستخرج الرقم النسبي للسجل ويحسب العنوان المادي ويقرأ السجل وتعمل أساليب فنية أخرى للنيل على حساب الرقم النسبي للسجل من مفتاح عددي.

توجد مشاكل بالنسبة لادارة المعطيات التقليدية. ولأن التطبيقات المختلفة تتطلب غالباً نفس المعطيات، فإن عناصر معينة من المعطيات تخزّن في أماكن عديدة مختلفة، ومن الصعب الحفاظ على مثل هذه المعطيات المتكرّرة. والمشكلة الأخرى هي الاعتماد على المعطيات. إذا كان منطق برنامج ما مرتبطاً بشكل وثيق بالبنية المادية للمعطيات، فإن الحفاظ على ذلك البرنامج يصبح صعباً. والحل للمشكلتين هو، غالباً، جمع معطيات المنظمة في قاعدة معطيات متمركزة. وبالعمل بنظام قاعدة المعطيات تكن هنالك نسخة واحدة فقط من كل عنصر مع معطيات، وهكذا يتم القضاء على تكرار المعطيات. ولأن كل برنامج يجب أن ينال المعطيات بواسطة نظام إدارة قاعدة معطيات، فإن البرامج تعزل عن البنية المادية للمعطيات، وبهذا الشكل يقلّل الاعتماد على المعطيات.

مصطلحات أساسية

- ☐ أسلوب النيل
- ☐ صفيف
- ☐ معطيات
- ☐ قاعدة معطيات
- ☐ نظام إدارة قاعدة معطيات
- ☐ تبعية المعطيات
- ☐ عنصر معطيات
- ☐ إدارة المعطيات
- ☐ بنية المعطيات
- ☐ النيل المباشر
- ☐ حقل
- ☐ ملف
- ☐ فهرس
- ☐ بنية لائحة
- ☐ دخل/خرج منطقي
- ☐ افتح
- ☐ دخل/خرج مادي
- ☐ نيل عشوائي
- ☐ سجل
- ☐ معطيات مكرّرة
- ☐ رقم سجل نسبي
- ☐ نيل تسلسلي

اختبار ذاتي

1. يحدّد نظام التشغيل موقع برنامج معين بتفتيش القرص بحثاً عن ____________

أ. مترجم ج. جدول
ب. تحميل ذاتي د. فهرس

2. أبسط بنية للمعطيات هي ____________.

أ. ملف ج. سجل
ب. لائحة د. صفيفة

3. تنال قيم المعطيات برقم (أو أرقام) عنصر في ____________.

أ. لائحة ج. ملف
ب. صفيف د. سجل

4. عنصر معطيات واحد له معنى، مثل إسم شخص، هو ____________.

أ. سجل ج. حقل
ب. عنصر د. ملف

5. مجموعة حقول ذات علاقة تشكل ____________.

أ. سجلا ج. صفيفا
ب. ملفا د. لائحة

6. مجموعة من سجلات ذات علاقة تشكل ____________.

أ. حقلا ج. لائحة
ب. صفيفا د. ملفا

7. يفكر المبرمج بلغة ____________.

أ. دخل/خرج المادية
ب. دخل/خرج المنطقية
ج. بدائية
د. أوامر

8. تحويل قطاع مفرد من قرص الى ذاكرة رئيسية هو مثال على:

أ. دخل/خرج
ب. دخل/خرج منطقي
ج. نيل ملف
د. بنية معطيات

9. إذا أعطينا رقم سجل نسبي، ______ يكون من الممكن أن نحسب عنوان القرص الـ______.

أ. منطقي/مادي
ب. مادي/منطقي
ج. حقيقي/زائف
د. زائف/حقيقي

10. طلبات المبرمج دخل/خرج المنطقية تحول الى أوامر مادية بواسطة ______.

أ. البرنامج
ب. الكيان المادي
ج. القناة
د. أسلوب النيل

11. تعالج السجلات بالترتيب الذي تسجل به في معالجة ______.

أ. مباشرة
ب. عشوائية
ج. تتابعية
د. مفهرسة

12. عندما يكون النيل ______ يمكن أن تنال السجلات بأي ترتيب.

أ. تتابعياً
ب. بالملف
ج. بالسجل
د. مباشراً

13. يحدث ______ المعطيات عندما تخزّن المعطيات نفسها في ملفين أو أكثر.

أ. إطناب
ب. نيل
ج. تبعية
د. فقدان

14. يحدث ______ المعطيات عندما يكون منطق البرنامج وثيق الصلة بالبنية المادي للمعطيات.

أ. إطناب
ب. نيل
ج. فقدان
د. تبعية

15. يمكن تجنّب إطناب المعطيات وتبعية المعطيات باستعمال ______.

أ. قاعدة المعطيات
ب. الملفات
ج. الملفات التتابعية
د. الملفات العشوائية

الاجابات

1.د 2.ب 3.ب 4.ج 5.أ 6.د 7.ب 8.أ 9.أ 10.د 11.ج 12.د 13.أ 14.د 15.أ.

ربط المفاهيم

1. إذا جرى نيل برنامج ما، فستنال جميع تعليماته. ويتم من الناحية الأخرى نيل المعطيات. إشرح.
2. ما هي بنية المعطيات؟ لمَ تكون بنى المعطيات مهمة؟
3. أوجد العلاقة بين رمز، حقل، سجل وملف.
4. ماذا يحدث عندما يفتح ملف؟ لماذا؟
5. إشرح بإيجاز الفرق بين دخل/خرج منطقي ودخل/خرج مادي.
6. إشرح بإيجاز مفهوم السجل النسبي
7. ميّز بين النيل التتابعي والنيل المباشر. أوجد العلاقة بين أسلوبي النيل ومفهوم السجل النسبي.
8. ما هو أسلوب النيل؟ لماذا تنشأ الحاجة لأساليب النيل؟
9. ما هي قاعدة المعطيات؟ قواعد المعطيات ـ لماذا هي مهمة؟
10. أوجد العلاقة بين نظام إدارة قاعدة المعطيات وأسلوب النيل. ما هو وجه الشبه بينهما؟ وما هي أوجه الاختلاف؟

9.

تصميم وتحليل الأنظمة

مفاهيم أساسية

الأنظمة

تحليل الأنظمة
- ☐ تحديد المشكلة
- ☐ التحليل
- ☐ التصميم
- ☐ التنفيذ
- ☐ الصيانة

الأنظمة

إن العمل تحت إشراف برنامج مخزون، يعالج الكمبيوتر المعطيات ليحوّلها الى معلومات. وإذا تأملت هذا التعريف للحظة، وجدت أن أي تطبيق لديك تتم معالجته عبر الكمبيوتر له على الأقل ثلاثة مكوّنات هي: الكيانات المادية والكيانات المنطقية والمعطيات. وإن مجرد كتابة برنامج لا يكفي لأن البرنامج ليس إلا أحد المكونات في نظام.

والنظام عبارة عن مجموعة مكونات تعمل معاً لتحقيق هدف ما. على سبيل المثال، إذا تفحّصنا نظام جدول الرواتب والذي يهدف الى دفع رواتب الموظفين، فما هي المكونات التي يتضمّنها النظام؟ يقوم الموظفون يومياً، بتسجيل ساعات العمل على بطاقات توقيت. وفي نهاية كل أسبوع، يتم جمع بطاقات التوقيت هذه وتقديمها الى مركز الكمبيوتر حيث تقرأ في برنامج جدول الرواتب. وخلال التنفيذ يقوم البرنامج بنيل معطيات الملفات. في النهاية يتم طبع شيكات الرواتب وتوزيعها على الموظفين. ولكي يعمل النظام، يجب التنسيق بحرص ما بين العوامل البشرية والاجراءات ووسائط الدخل والخرج والملفات والكيانات المادية والمنطقية. ويجب الانتباه الى أن البرنامج ليس إلا أحد المكونات في نظام ما.

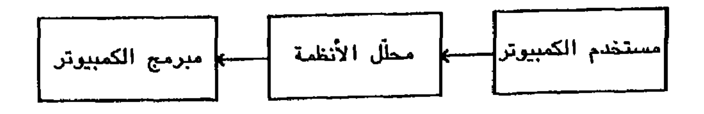
تحليل الأنظمة

يتم تطوير الأنظمة القائمة على الكمبيوتر نظراً لحاجة الناس الى المعلومات. إن هؤلاء الناس الذين يسمون بشكل عام **مستخدمي الكمبيوتر** على علم بما هو مطلوب ولكن قد تنقصهم الخبرة للحصول عليه. أمّا المهنيون التقنيون في حقل الكمبيوتر قد يكونون حائزين على الخبرة ولكنهم قد يفتقرون الى التمرين الذي يكتسبه مستخدمو أجهزة الكمبيوتر. ولتعقيد الأمور يبدو في الغالب، أن مستخدمي ومبرمجي الكمبيوتر يتحدثون بلغات مختلفة مما يؤدّي الى حدوث مشاكل في الاتصال فيما بينهم. وإن **محلّل الأنظمة** هو المهني الذي يقوم بتفسير حاجات مستخدم الكمبيوتر الى مصطلحات فنية (شكل 9.1) وهو بذلك يقوم مقام صلة وصل ما بين مستخدمي الكمبيوتر من جهة والمهنيين التقنيين من جهة أخرى.

يعمل محلّل الأنظمة كالمهندس أو المهندس المعماري، على حل المشاكل عن طريق التوفيق بين الخبرات الفنية المحسوسة ونفاذ البصيرة وحسن التصوّر والقدرة على إعطاء لمسة فنية للأمور. ويتبع عامة، طريقة معالجة منهجية محدّدة جيداً تشمل ما لا يقل عن الخطوات التالية:

1. تحديد المشكلة.
2. التحليل.
3. التصميم.
4. التنفيذ.
5. الصيانة.

الشكل 9.1 يقوم محلّل الأنظمة بتفسير حاجات مستخدم الكمبيوتر الى مصطلحات فنية.



وعند نهاية كل خطوة من هذه الخطوات، يتم توثيق النتائج وتقاسمها مع كل من مستخدمي الأنظمة ومبرمجيها. والغاية من هذا هي كشف الأخطاء وأسباب سوء التفاهم والعمل على تصحيحها في أقرب وقت ممكن. وقد يكون المثل الوسيلة الفضلى لايضاح طريقة المعالجة.

لنتصوّر مخزناً صغيراً للألبسة يتم فيه شراء البضائع بالجملة وعرضها ثم بيعها بالمفرق الى الزبائن. فمن ناحية، يمثل شراء كمية كبيرة من البضائع مرة واحدة تكاليف لا ضرورة لها. ومن ناحية أخرى، إن عرض كمية قليلة من البضائع للبيع لا يشجع المتسوّقين على الشراء نظراً لضيق مجال الاختيار. وبصورة مثالية، يمكن تحقيق التوازن بين هذين الأمرين ولكن ليس بالقدر الكافي.

إن تعقيد الأمور في الواقع ناتج عن عملية جرد البضائع التي تتغير باستمرار عن طريق نفاذ الكمية المعروضة للبيع نتيجة لمشتريات الزبائن، وعن طريق البضائع التي تُعاد والطلبات الجديدة التي تُضاف الى هذه الكمية. ويكون المالك راغباً بمتابعة مستويات جرد البضائع نفسها وبإعداد طلبات جديدة على أي صنف معين قبل نفاذ الكمية الموجودة لديه في المخزن بقليل. وقد تكون المهمة في منتهى السهولة بالنسبة لصنف واحد حيث يتمكن البائع أن يعد ما هو موجود من هذا الصنف في المخزن. لكن لسوء الحظ هناك مئات من الأصناف المختلفة في المخزن ومتابعة أثر كل منها شيء غير عملي. هنا قد يساعد الكمبيوتر على حل المشكلة.

تحديد المشكلة

إن الخطوة الأولى في عملية تصميم وتحليل الأنظمة هي **تحديد المشكلة**. إن هدف المحلّل هو أن يتحقّق مما يحتاجه مستخدم الكمبيوتر (في هذه الحالة، صاحب المخزن). يجب أن نلاحظ عند بدء العملية أن لدى مستخدم الكمبيوتر المعلومات المهمة وعلى المحلّل أن يصغي الى ويتعلم هذه المعلومات. إن قلة من مستخدمي الكمبيوتر هم خبراء فنيون والأغلبية تعتبر جهاز الكمبيوتر «كصندوق سحري» ولا يبدون اهتماماً بكيفية عمله. في هذه المرحلة، لا يبدي المحلّل اهتماماً ولا حتى تفكيراً بالبرامج والملفات والكيانات المادية لجهاز الكمبيوتر بل يقع على عاتقه إجراء الاتصال مع مستخدم الكمبيوتر وذلك بالأسلوب الذي يتناسب معه رجلاً كان أو امرأة.

فالفكرة هي أن نتأكد من أن كلي الشخصين، مستخدم الكمبيوتر ومحلّل الأنظمة يفكران بالشيء نفسه. لذلك ندرك بأن إصدار تصريح كتابي واضح يعبر عن تفهّم المحلل للمشكلة هو أمر ضروري. وعلى مستخدم الكمبيوتر أن يراجع ويصحح هذا التصريح. والآن حان الوقت لكشف المواقع التي يساء فهمها ولتحديد السهو قبل ضياع الوقت والمال والجهد.

في الغالب، على أثر تحديد مشكلة تمهيدية، يقوم محلل الأنظمة بإجراء **دراسة جدوى** تكون بمثابة نسخة موجزة عن عملية تصميم وتحليل الأنظمة بكاملها، وترمي هذه الدراسة الى الاجابة على ثلاثة أسئلة:

1. هل من الممكن إيجاد الحل للمشكلة؟
2. هل من الممكن إيجاد الحل للمشكلة ضمن نطاق مفاهيم مستخدم الكمبيوتر؟
3. هل من الممكن إيجاد حل للمشكلة بكلفة معقولة؟

إن كانت الاجابة على أي من هذه الأسئلة بالنفي يجب أن لا يتم تطوير النظام. ويتوفّر تحديد جيد للمشاكل ودراسة جدوى إيجابية يستطيع المحلّل أن يوجّه اهتمامه نحو التخطيط والتطوير لحل هذه المشاكل.

التحليل

عندما يبدأ التحليل يتفهم المحلل المشكلة، ويتقرّر في الخطوة التالية ما يجب القيام به لحلها. فالمستخدم يعرف ما يجب عليه أن يفعل. فتستخلص هذه المعرفة وتوثق رسمياً في أثناء التحليل. ويفكر معظم مستخدمي أجهزة الكمبيوتر بلغة الوظائف التي يجب إنجازها ومن خلال عناصر المعطيات التي يجب معالجتها. والهدف من ذلك هو التعرف على وربط هذه الوظائف الأساسية وعناصر المعطيات ينتج عنه تصميم **لنظام منطقي**.

لنبدأ بوظائف النظام الأساسية. فالمفتاح الرئيسي يتتبع آثار البضاعة الموجودة في المخزن لأن كل صنف إنتاج مدوّن في لائحة الموجودات. إن عملية جرد المبيعات تتغير نتيجة لمشتريات الزبائن وللمبادلة، ونتيجة لاعادة بضائع مباعة لذلك يجب أن يعالج النظام معاملات الزبائن التجارية. ثم أن صاحب المخزن يود أن يتفحّص مستوى لائحة الموجودات لأي نوع إنتاج تقل كمية مخزونها من البضائع، وأن يتقدّم بطلب كمية بديلة منها إذا كان ذلك مناسباً، لذلك يجب أن يتمكّن النظام من الاتصال بإدارة العمل. وأخيراً، بعد الحصول على تفويض إداري، يجب على النظام أن يتقدّم بطلب متكرّر ويكون على استعداد لارساله الى منتج البضائع.

عندما تتوفّر وظائف النظام الأساسية تكون المهمة التالية للمحلل أن يتمكن من إجراء ترابط منطقي بينها. والوسيلة الصالحة للبدء بهذه المهمة هي عن طريق وصف كيفية تدفق المعطيات بين الوظائف. وما مخطّطات **تدفق المعطيات** إلا التسمية التي تدل على فائدة خاصة ألا وهي وصف هذا التدفق بشكل بياني. وتستخدم في هذا الوصف أربعة رموز (الشكل 9.2). فمصادر المعطيات ومقصدها تتمثل بمربعات، وتدخل معطيات الدخل الى النظام من مصدر وتتدفّق معطيات الخرج نحو مقصد. وعند دخول المعطيات في نظام يتم التداول بها أو تغييرها بواسطة عمليات معالجة تتمثل بمستطيلات مستديرة الزوايا. وقد تكون عملية المعالجة برنامجاً واجراءً معيناً أو أي شيء آخر يغير أو يحرّك المعطيات. ويمكن حفظ هذه المعطيات في مواضع خزن لمعالجتها فيما بعد، وتكون هذه المواضع ممثلة بمستطيلات مفتوحة الأطراف. وقد يكون موضع خزن المعطيات في ملف قرص، وفي ملف شريط، وفي قاعدة معطيات، وفي ملاحظات مكتوبة، أو حتى في ذاكرة الانسان. وفي النهاية، تتدفق المعطيات بين المصادر والمقاصد وعمليات المعالجة ومواضع خزن المعطيات على خطوط تدفّق المعطيات الممثلة بسهام.

الشكل 9.2 يستخدم رسم تدفق المعطيات أربعة رموز لكي يمثل مكونات نظام منطقي.

مصدر او وجهة معطيات

عملية معالجة نقل المعطيات

موضع خزن المعطيات

تدفق المعطيات

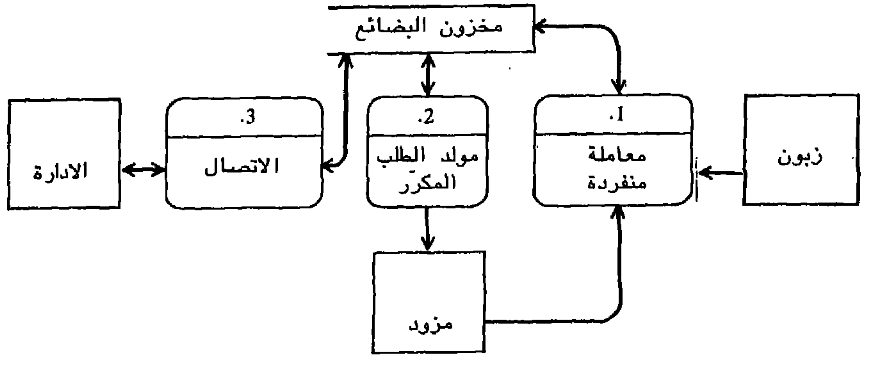
يبيّن الشكل 9.3 رسماً لتدفّق المعطيات التمهيدية لنظام لائحة الموجودات. وعندما تبدأ بكلمة «زبون» تتدفق المعاملات التجارية منها الى النظام حيث يتم التعامل بها بشكل معاملة منفردة. ويحفظ مخزن المعطيات، أي مخزون من البضائع، المعطيات بشأن كل صنف من المبيعات وتغير المعاملة المنفردة المعطيات لتعكس معاملة جديدة. في تلك الأثناء، تقوم الادارة بنيل النظام عبر الاتصال وتقوم بتقييم المعطيات الموجودة في مخزون البضائع. وإذا لزم الأمر، يقدم طلباً مكروراً. وحالما يتم التفويض بتقديم الطلب المكرّر يرسل مولد الطلب المكرر المعطيات اللازمة الى المنتج الذي يقوم بشحن الأصناف المطلوبة من البضائع الى المخزن. عليك أن تلاحظ أنه نظراً للتغير الذي يمثله الطلب المكرّر في مستوى قائمة المبيعات لمادة أو مواد معينة، فإن هذه المادة أو المواد تعالج كمعاملة.

يصف رسم تدفق المعطيات النظام المنطقي للكمبيوتر. وتكون الخطوة التالية تتبّع اتجاهات تدفق المعطيات. وعليك أن تبدأ من مقصد المنتج. وإن أجراءات الطلب المكرّر تتدفق باتجاه المنتجين، على سبيل المثال، قد يكون المخزن بحاجة لـ25 سروالاً. لملء هذا الطلب يحتاج المنتج الى وصف لهذه المادة والكمية المتكرّر طلبها. من أين لنا أن نجد عناصر المعطيات هذه؟ وبما أنها تخرج بواسطة مولد الطلب المكرّر يجب أن تكون قد ادخلت أو ولدت من هذه العملية. تتدفق المعطيات الى مولد الطلب المكرّر من مخزون البضائع لذلك يجب أن تكون أوصاف المادة المطلوبة وكمياتها المتكرّر طلبها مخزونة في مخزون البضائع.

يتم توليد عناصر المعطيات الأخرى مثل النفقات والكميات التي تمّ شراؤها بواسطة عملية الزبون. وهناك عناصر أخرى أيضاً مثل، سعر المبيع وإشارة الطلب المكرّر يتم توليدها أو احتياجها من قبل الادارة. ويمثل مخزون البضائع الحالي لصنف معين نموذجاً عن عناصر المعطيات التي يتمّ توليدها بواسطة لوغارتم في واحد من الاجراءات. وبأسلوب موضوعي، وفي كل خطوة على حدة يقوم المحلل بتمييز عناصر المعطيات التي يجب أن تدخل النظام، أو تخزّن، أو تعالج فيه، أو تولد، أو تخرج منه.

لتتبع عناصر المعطيات، يدرج محلل الأنظمة كل عنصر في قاموس معطيات (الشكل 9.4). ويمكن تشكيل قاموس معطيات بسيط على بطاقات فهرس، ولكن قواميس المعطيات التي تجهز بها الكمبيوترات أصبحت شائعة الاستخدام بشكل متزايد. وإن لقاموس المعطيات الذي هو عبارة عن مجموعة معلومات تصف وتحدد المعطيات فائدته خلال كل عملية تصميم وتحليل الأنظمة، ويستخدم في الغالب، لبناء قاعدة معطيات خلال مرحلة التنفيذ.

الشكل 9.3 رسم تدفق المعطيات لنظام مبيعات.



الشكل 9.4 يتمّ توثيق عناصر المعطيات رسمياً في قاموس معطيات.

الاسم:	وصف المادة.
التعريف:	الوصف.
الوصف:	وصف شفهي موجز للمادة مثل صنفها وحجمها ولونها، الخ.
النسق:	أبجدي.

الاسم:	المنتج.
التعريف:	اسم المنتج، البائع.
الوصف:	اسم الشركة التي تزوّد المخزن بالمنتجات.

الاسم:	الموجودات من مخزون البضائع.
التعريف:	مخزون البضائع، الكمية.
الوصف:	عدد الوحدات من مادة معينة موجودة حالياً في لائحة جرد الموجودات.

الاسم:	الكمية المكرّر طلبها.
التعريف:	لا شيء.
الوصف:	عدد الوحدات من مادة معينة يجب أن يُعاد طلبها في كل مرة تدعو الحاجة الى مخزون جديد من البضائع.
النسق:	عددي.
الموقع:	مخزون البضائع.

الفكرة من التحليل هي تحديد الوظائف الرئيسية للنظام وعناصر المعطيات بطريقة منهجية. ويجب التذكر أن الهدف هو تفسير ما يحتاج اليه مستخدم الكمبيوتر في مصطلحات فنية. وبما أن النظام يبدأ مع المستخدم، فالخطوة الأولى تحدّد ما يحتاج إليه هذا المستخدم. وأمّا طريقة تفكير مستخدمي أجهزة الكمبيوتر فتتم ضمن نطاق الوظائف والمعطيات. كما أنه ليس لدى هؤلاء المستخدمين تصوراً للبرامج أو الملفات أو للكيانات المادية. ففي خلال مرحلة التحليل هذه الدقيقة والأساسية، من الضروري جداً للمحلّل أن يفكر كما يفكر مستخدم الكمبيوتر لا كما يفكر المبرمج.

تشكّل رسوم تدفق المعطيات وقواميس المعطيات أدوات لها فائدتها في أنظمة الكمبيوتر، فهي تعطي نسقاً لتسجيل المعلومات الرئيسية بشأن نظام مقترح. كما أنها تنبّه ذاكرة محلل الأنظمة، على سبيل المثال، إذا لم يكن لدى المحلل المعلومات الكافية لاتمام إدخال معطيات في القاموس فمن المحتمل أن يكون قد فاته، أو فاتها شيء ما. والأمر الذي قد يكون ذو أهمية بالغة أن رسم تدفّق المعطيات وقاموس المعطيات يعملان على توثيق مفهوم المحلّل لمتطلبات النظام. وبمراجعة هذه الوثائق، يستطيع مستخدم الكمبيوتر أن يقوم بتصحيح ما ساء فهمه أو بتصحيح الأخطاء التي وقعت سهواً. وفي النهاية، تمثّل هذه الأدوات نقطة بداية ممتازة للخطوة القادمة وهي التصميم.

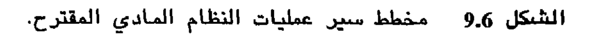
التصميم

بدخولنا مرحلة **التصميم** نعرف ما يجب على النظام أن يفعل وهكذا نستطيع أن نبدأ التفكير بكيفية القيام بذلك. والهدف هو أن نطوّر أسلوباً استراتيجياً لحل المشكلة. في هذه المرحلة، لا يهمنا أن نكتب كوداً أو أن نحدّد بنى دقيقة للمعطيات، وما يهمّنا أننا نريد أن نعين البرامج اللازمة والملفات والاجراءات والمكونات الأخرى على مستوى الصندوق الأسود.

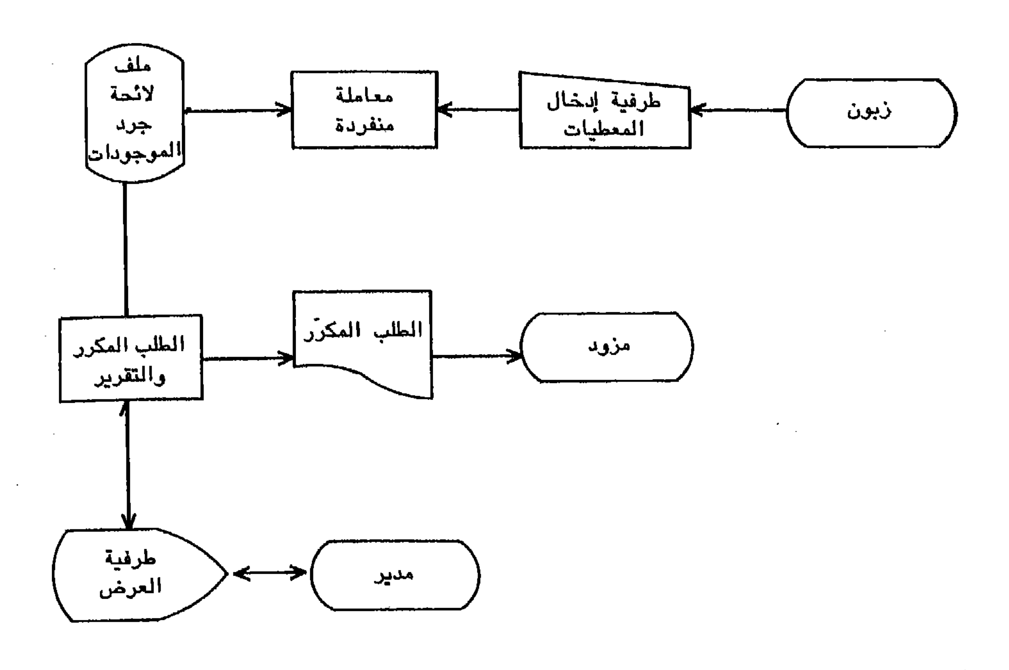
يحدّد رسم تدفّق المعطيات وظائف النظام الضرورية، فكيف يمكن تنفيذ هذه الوظائف؟ إحدى الامكانيات هي كتابة برنامج لكل عملية معالجة والأخرى هي جمع عمليتَي معالجة أو أكثر في برنامج واحد، وهناك الكثير من الحلول البديلة. فلنركّز على خيار واحد ونقوم بتوثيقه.

يستخدم **مخطط سير عمليات النظام** رموزاً لتمثيل برامج وإجراءات وأجهزة كيانات مادية ومكونات أخرى **لنظام مادي** (الشكل 9.5). يبيّن مخطط سير عملياتنا (الشكل 9.6) أن معاملة المعطيات تدخل النظام عبر طرفية، وتعالج بواسطة برنامج جمع المعطيات ثم تخزّن على ملف لائحة جرد الموجودات. وأخيراً، تتم معالجة ملف لائحة جرد الموجودات بواسطة برنامج طلب مكرّر وتقرير. ومن خلال هذا البرنامج، تتداول الادارة أمر المعطيات وتصدر تفويضاً بإجراء بالطلبات المكرّرة.

الشكل 9.5 على مخطط سير عمليات نظام، تمثل الرموز برامج وإجراءات وأجهزة كيانات مادية ومكوّنات أخرى لنظام مادي.



الشكل 9.6 مخطط سير عمليات النظام المادي المقترح.



إذا نظرت الى مخطط سير عمليات النظام تجد انه يعين عدة مكونات كيانات مادية من ضمنها جهاز كمبيوتر ومدوار أقراص وطرفية ادخال معطيات وطابعة وطرفية عرض. وهنا تدعو الحاجة الى برنامجين: برنامج عملية منفردة وبرنامج طلب مكرّر وتقرير. وبالاضافة الى الكيانات المادية والبرامج، سنحتاج الى بنى معطيات لملف لائحة جرد الموجودات ولخطوط تدفق معطيات بين أجهزة الدخل/الخرج والكيانات المادية. والجدير بالملاحظة أن مخطط سير عمليات النظام هذا يوضح إمكانية إيجاد حل واحد مع العلم أن محلل الأنظمة الجيد سيقوم بتطوير عدة حلول بديلة مجدية قبل أن يختار حلاً واحداً.

يرسم مخطط سير العمليات النظام مبرزاً مكوّناته المادية الرئيسية. وبما أن المعطيات تتصل بالمكونات، فتكون المهمة التالية تحديد بنى المعطيات. على سبيل المثال، إذا أخذت بالاعتبار ملف لائحة جرد الموجودات تجد أنه يحتوي على كل عناصر المعطيات من معطيات مخزون البضائع في المخزن. وتكون عناصر المعطيات مدرجة في قاموس المعطيات. فباستخدام هذه العناصر يمكن تخطيط بنى المعطيات للملف.

كيف يجب تنظيم الملف؟ إن ذلك يعتمد على كيفية نيله. على سبيل المثال، تتمّ معالجة المعطيات في بعض التطبيقات على فترات يمكن التكهّن بها ومنتظمة. وبصورة نموذجية، يتم جمع المعطيات على فترة زمنية، ومعالجتها معاً كدفعة واحدة. وإذا كانت **المعالجة بدفعات** مقبولة، فقد يكون تنظيم الملفات التسلسلي هو الأفضل.

مع ذلك، لا يمكن الانتظار دائماً حتى يتم جمع دفعة من المعاملات. على سبيل المثال، لنأخذ بالاعتبار نظام تحذير مبكر في الدفاع الجوي، إذا تمّ كشف طائرة حربية غير محدّدة الهوية، فيجب تحديد هويتها على الفور. وفكرة الانتظار حتى الساعة الخامسة مساء، موعد «تنفيذ برنامج الدفاع الجوي»، هو أمر مناف للعقل. بدلاً من ذلك، ومن منطلق الحاجة للقيام برد فعل سريع، يتوجب معالجة كل معاملة وفقاً للظروف التي تحدث فيها. وبصورة عامة، تتطلب أنظمة **معالجة المعاملات** هذه ملفات نيل مباشر.

لنظامنا الخاص بلائحة جرد الموجودات برنامجان يقوم أحدهما بمعالجة المعاملات. ويبدو لنا برنامج ملف مبيعات نيل مباشر خياراً معقولاً. كما يسمح البرنامج الآخر للادارة بدراسة معطيات المبيعات من وقت الى آخر، ومما لا شك فيه أنه يمكن استخدام المعالجة بدفعات. هل يجب تنظيم ملف المبيعات بطريقة تسلسلية أو بطريقة مباشرة؟ فعندما يواجهنا خيار كهذا، يأخذ محلل الأنظمة الجيد بالاعتبار الخيارين معاً. وقد يقبل نظام ما المعاملات ويعالجها بالحالة التي ترد فيها. وفي خطوة بديلة، يمكن جمع أوراق المبيعات التي تتم خلال اليوم ومعالجتها بدفعات بعد إغلاق المخزن. في النظام الأول، يتعامل البرنامجان مع ملفات نيل مباشر. وفي النظام الثاني، يتم توصيلهما الى ملفات تسلسلية. والبرنامج الذي يعالج معطيات نيل مباشر يختلف عن البرنامج الذي يعالج معطيات تسلسلية. والمعطيات هي التي تدير النظام. أمّا اختيار بنية المعطيات هو الذي يحدّد بنية البرنامج. كما يجب أن تلاحظ أن البرنامج محدّد ومخطط له ضمن سياق النظام.

التنفيذ

ما إن يتم تعيين المكوّنات الرئيسية للنظام، نستطيع البدء بتطوير هذه المكوّنات. ويحتوي نظامنا على برنامجين، وعلى عدة قطع من المعدات، وعلى عدد من بني المعطيات. ويكون خلال **التنفيذ**، كل برنامج مخططاً ومكتوباً عن طريق استخدام الأساليب الفنية التي ورد وصفها في الفصل 7. ويتم تشكيل الملفات وتدقيق محتوياتها. كما يتم شراء كيانات مادية جديدة وتركيبها واختبارها. بالاضافة الى ذلك، تكون إجراءات التشغيل مكتوبة ومقيمة. وعندما يتمّ تحضير كل أجزاء المكونات، يجري اختباراً للنظام. وإذا افترضنا أن مستخدم الكمبيوتر راض عن النتائج يعتبر النظام الذي تم الانتهاء من تصميمه قابلاً للاستخدام.

الصيانة

تبدأ **الصيانة** بعد أن يصبح النظام قابلاً للاستخدام. وعندما يبدأ الناس باستخدامه يقترحون إجراء تحسينات وتطويرات بسيطة عليه. وفي بعض الأحيان، تنسل بعض الشوائب في النظام عبر الاختبار وكشف الخلل وتصحيحه، وتعتبر إزالتها مهمة صيانة أخرى. في النهاية، تتغير الظروف مما يوجب إجراء تحديث على البرامج. على سبيل المثال، إذا أقرّت الحكومة قانوناً يقضي بتغيير إجراء جمع ضرائب الدخل فيجب أن يتم تعديل برنامج دفع الرواتب. وتستمر الصيانة طوال مدة عمل النظام ويمكن لتكاليفها أن تضاهي بسهولة أو تتجاوز تكاليف تطوير النظام الأولية. ويمكن أن يساعد التخطيط الجيد والتوثيق السليم والبرامج المتينة البنية في تخفيض تكاليف الصيانة الى الحد الأدنى.

الخلاصة

النظام هو مجموعة من الكيانات المادية والمنطقية والمعطيات ومكونات الاجراءات التي تعمل معاً لتحقّق هدفاً معيناً. والبرنامج هو مجرد مكون واحد في نظام.

يتم تخطيط الأنظمة وتصميمها من قبل محللي الأنظمة الذين يتبعون بصورة عامة، عمليات معالجة منهجية ومحدّدة جيداً. والخطوة الأولى في هذه العمليات تكون تحديد المشاكل هذا عندما يحاول المحلل أن يكتشف بالضبط ما يحتاج اليه مستخدم الكمبيوتر.

فغالباً، وعلى أثر القيام بتحديد مسبق للمشاكل، يتم إجراء دراسة جدوى للتحقّق من إمكانية إيجاد حل للمشكلة.

بعد الحصول على تحديد واضح للمشاكل، يبدأ التحليل. خلال هذه المرحلة، يطوّر محلل الأنظمة نموذجاً منطقياً للنظام. ويتم توصيل الوظائف الرئيسية للنظام عن طريق رسوم تدفق المعطيات. وباستخدام هذه الرسوم كأدوات، يتم إتباع خطوط تدفق المعطيات ويتم تعيين عناصر معطيات النظام وتسجيلها في قاموس معطيات. وبعد مراجعة النظام المنطقي مع مستخدم الكمبيوتر، يبدأ التصميم.

خلال التصميم، يطوّر محلل الأنظمة نموذجاً عن النظام المادي ويمكن استخدام مخطط سير عمليات النظام لرسم النظام المطلوب مع تحديد كل مكون مادي كرمز. ويأخذ محلل الأنظمة الجيد بالاعتبار عدداً من الحلول البديلة للمشاكل قبل أن يقرّر استخدام حلّ واحد. ويتبع التنفيذ عملية التصميم. فتكون البرامج قد خطّطت وكتبت، والكيانات المادية قد طلبت وركبت، والاجراءات قد كتبت، والملفات وقواعد المعطيات قد أسّست، وقطع المعدات قد جمعت واختبرت في النهاية. وبعد أن يصبح النظام قابلاً للاستخدام، تبدأ الصيانة.

مصطلحات أساسية

☐ تحليل	☐ تنفيذ	☐ مخطط سير عمليات النظام
☐ معالجة دفعية	☐ نظام منطقي	☐ تحليل الأنظمة
☐ قاموس المعطيات	☐ الصيانة	☐ محلّل الأنظمة
☐ رسم تدفق المعطيات	☐ نظام مادي	☐ معالجة المعاملات
☐ التصميم	☐ تحديد المشاكل	☐ مستخدم
☐ دراسة جدوى	☐ نظام	

اختبار ذاتي

1. الـ__________ هو مكوّن واحد في __________.

أ. نظام/برنامج
ب. برنامج/نظام
ج. نظام/كمبيوتر
د. كمبيوتر/برنامج

2. المهني الذي يفسّر ما يحتاج اليه مستخدم الكمبيوتر في مصطلحات فنية هو __________.

أ. المستخدم
ب. محلّل الأنظمة
ج. المبرمج
د. المدير

3. الخطوة الأولى في عملية تصميم وتحليل الأنظمة هي __________.

أ. دراسة جدوى
ب. التحليل
ج. تحديد المشاكل
د. التصميم

4. خلال عملية تحديد المشاكل، يكون __________ مصدراً للمعلومات الدقيقة.

أ. المستخدم
ب. محلّل الأنظمة
ج. المبرمج
د. المدير

5. قد يقوم محلل الأنظمة بإجراء ________ ليتحقّق من إمكانية إيجاد حل لمشكلة.

أ. تحليل
ب. تصميم
ج. دراسة جدوى
د. تحديد مشاكل

6. خلال التحليل، يقوم المحلّل بتخطيط ________.

أ. نظام
ب. نظام مادي
ج. نظام منطقي
د. دراسة جدوى

7. يمكن توثيق تصميم النظام المنطقي بواسطة ________.

أ. مخطط سير عمليات
ب. رسم تدفق المعطيات
ج. مخطط سير عمليات النظام
د. قاموس المعطيات

8. خلال عملية التحليل، يكون هدف محلل الأنظمة أن يعيّن ________ للنظام.

أ. الوظائف المنطقية
ب. عناصر المعطيات
ج. الاثنان معاً
د. لا شيء منهما

9. لتتبّع عناصر المعطيات، قد يقوم محلّل الأنظمة بتطوير ________.

أ. قاموس معطيات
ب. ملف
ج. رسم تدفق معطيات
د. مخطط سير عمليات النظام

10. يبدأ الاهتمام بالبنية المادية لنظام خلال ________.

أ. التحليل
ب. التصميم
ج. التنفيذ
د. تحديد المشاكل

11. يمكن رسم تصميم نظام مادي في ________.

أ. مخطط سير عمليات
ب. مخطط سير عمليات النظام
ج. رسم تدفق المعطيات
د. قاموس المعطيات

12. أي تنظيم ملفات تختاره لتطبيق معالجة دفعية؟

أ. مباشر
ب. عشوائي
ج. مفهرس
د. تسلسلي

13. أي تنظيم ملفات تختاره لتطبيق معالجة معاملات؟

أ. مباشر
ب. تسلسلي
ج. أحدهما
د. لا شيء منهما

14. تتم كتابة البرامج خلال ______________.

أ. التنفيذ ج. الصيانة
ب. التصميم د. التحليل

15. بعد «الانتهاء» من تهيئة النظام للاستخدام، يبدأ ______________.

أ. التنفيذ ج. القيام بالصيانة
ب. كشف الخلل وتصحيحه د. التوثيق

الاجابات

1.ب 2.ب 3.ج 4.أ 5.ج 6.ج 7.ب 8.ج 9.أ 10.ب 11.ب 12.د 13.أ 14.أ 15.ج.

ربط المفاهيم

1. ما هو النظام؟
2. يتم تحديد البرامج وتخطيطها ضمن سياق النظام. إشرح ذلك.
3. ما هي وظيفة محلل الأنظمة؟ ولماذا هناك حاجة اليه؟
4. ضع في قائمة الخطوات التي تتم في عملية تصميم وتحليل الأنظمة واشرح بإيجاز ما يحدث في أثناء كل خطوة منها.
5. في أثناء عملية تحديد المشاكل، يتوجّب على محلل الأنظمة أن يجري الاتصال مع مستخدم الكمبيوتر بالأسلوب الذي يفهمه المستخدم. لماذا؟
6. ما هي دراسة الجدوي؟ وما الذي يسعى الى تحقيقه محلّل الأنظمة من خلال هذه الدراسة؟
7. إشرح بإيجاز الفرق بين نظام منطقي ونظام مادي.
8. يتبع محلّل الأنظمة عملية منهجية تتم خطوة خطوة عند قيامه بتطوير نظام الكمبيوتر. لماذا؟
9. ميّز بين المعالجة الدفعية ومعالجة المعاملات.
10. يأخذ محلّل الأنظمة الجيد بالاعتبار عدة حلول بديلة لتصاميم أنظمة مادية قبل أن يختار حلاً واحداً. لماذا؟

10.

البرمجة والمعالجة المتعدّدتان

مفاهيم أساسية

البرمجة المتعددة

أنظمة التشغيل ذات البرمجة المتعددة

- □ إدارة وقت المعالج
- □ إدارة الذاكرة
- □ تخصيص أجهزة الدخل/الخرج
- □ الجدولة
- □ الخزن المؤقت السريع

المشاركة الزمنية

المعالجة المتعددة

البرمجة المتعددة

كانت الكمبيوترات القديمة قادرة على تنفيذ بضعة آلاف من التعليمات في الثانية، أمّا الكمبيوترات الرئيسية الحديثة فهي أسرع بكثير، إذ يمكنها أن تنفّذ ملايين التعليمات في الثانية. ومن الصعب تخيّل هذه السرعات، لذا يكفي أن نقول أن كمبيوترات اليوم هي حقاً سريعة جداً.

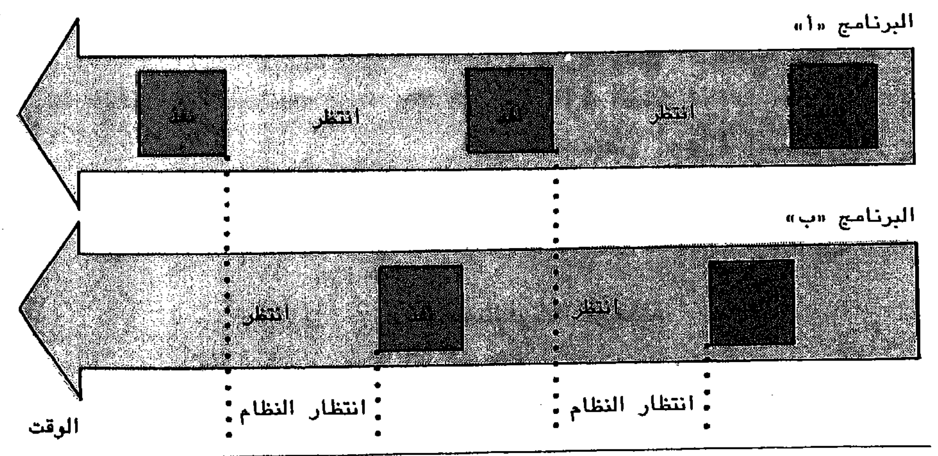
لسوء الحظ لم تجارِ سرعة الأجهزة المحيطية هذا الركب، فلوحات المفاتيح يشغلها البشر، ما مدى سرعتك في الضرب على الآلة الكاتبة؟ إن قارئات البطاقات والطابعات والأجهزة المماثلة ترسل بضعة آلاف من السمات في الثانية في أفضل الأحوال، حتى أن الخزن المساعد «الفائق السرعة» يعتبر نداً ضعيفاً إذا قورن بالكمبيوتر. ويستطيع الكمبيوتر الرئيسي الحديث معالجة المعطيات بسرعة تزيد مئات المرات - وحتى آلاف المرات - عن السرعة التي تستطيع الأجهزة المحيطية أن تجهّزها بها.

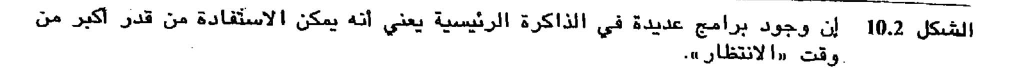
ماذا يفعل الكمبيوتر في أثناء الدخل أو الخرج؟ لا شيء. فالبرنامج لا يستطيع أن يعالج معطيات لم يحصل عليها بعد، ولا يمكن افتراض نجاح عملية خرج قبل أن تنتهي العملية، وهكذا فإن البرنامج ينتظر، وبما أن البرنامج يتحكّم بالكمبيوتر فإن الكمبيوتر ينتظر أيضاً.

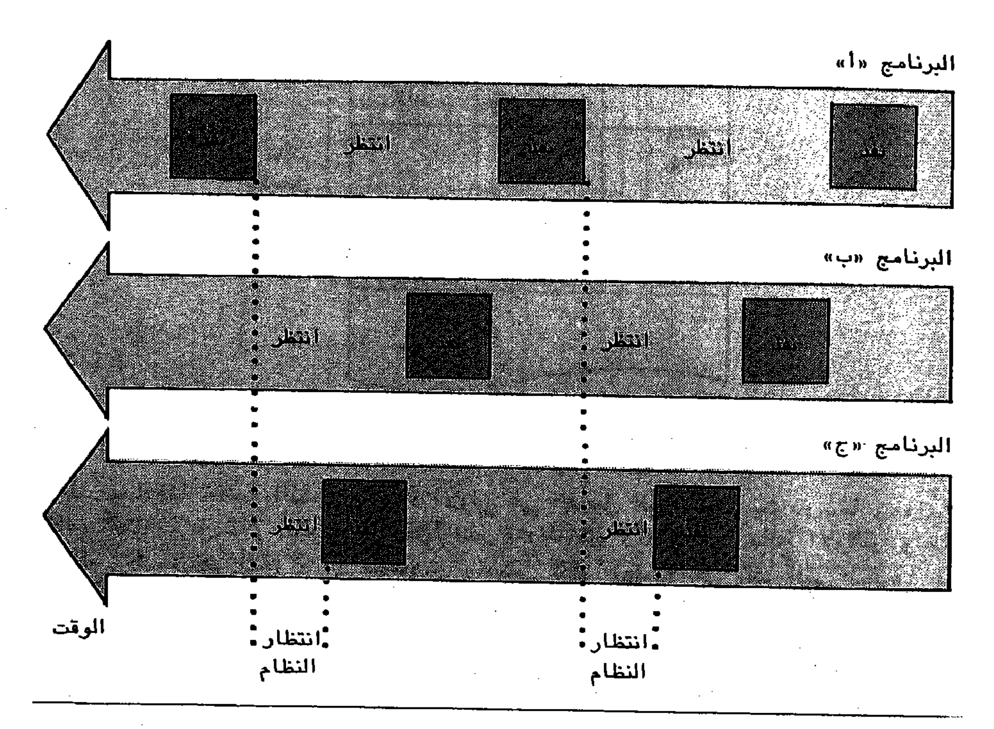
وبما أنه يفوق أجهزته المحيطية سرعة يقضي الكمبيوتر في انتظار الدخل والخرج عادة وقتاً أطول بكثير مما يصرفه في معالجة المعطيات لأنه أسرع بقدر كبير جداً من أجهزته المحيطية. فخلال ثانية واحدة يستطيع كمبيوتر رئيسي ضخم أن ينفذ مليون تعليمة أو أكثر، وهكذا تمثل كل ثانية لا تستغل هدراً هائلاً لطاقة الكمبيوتر المحتملة. والمشكلة تشبه الى حد ما تسيير قطار فائق السرعة على سكك رديئة الصيانة، فما هو النفع من سرعة لا تستطيع أن تستخدمها؟

لماذا لا نضع برنامجين في الذاكرة الرئيسية؟ حينئذ، وفي أثناء انتظار البرنامج «أ» للمعطيات، يستطيع المعالج أن يولي اهتمامه للبرنامج «ب» (الشكل 10.1). ولماذا نتوقف عند برنامجين؟ أمّا باستعمال ثلاثة برامج أو أكثر فيمكن الاستفادة من الوقت الذي فيما عدا ذلك يضيع سدى (الشكل 10.2). وبصورة عامة فإنه كلما ازداد عدد البرامج في الذاكرة كلما كانت الاستفادة من المعالج أكبر، ويُسمّى هذا الأسلوب **بالبرمجة المتعددة**.

الشكل 10.1 مع وجود برنامجين في الذاكرة الرئيسية يمكن للمعالج ان ينقل اهتمامه إلى البرنامج «ب» عندما يكون البرنامج «أ» في انتظار الدخل او الخرج.




الشكل 10.2 إن وجود برامج عديدة في الذاكرة الرئيسية يعني أنه يمكن الاستفادة من قدر أكبر من وقت «الانتظار».

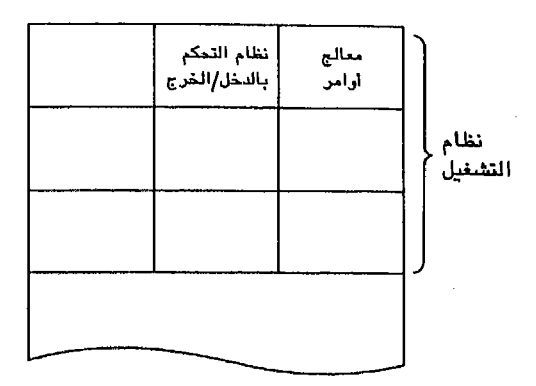


لاحظ أن المعالج لا يعمل إلّا على برنامج واحد فقط بالرغم من وجود عدة برامج مختلفة في الذاكرة الرئيسية. لماذا؟ لأن المعالج يستحضر تعليمة واحدة أثناء دورة الآلة الواحدة وينفذها، وإذا كان المعالج لا يستطيع أن ينفذ إلاّ تعليمة واحدة في وقت واحد فليس من المحتمل أن يكون قادراً على تنفيذ أكثر من برنامج واحد في آن، إن كلمة متواقت تعني «في اللحظة عينها». وما من أحد يستطيع أن يدرس ويشاهد التلفزيون في الوقت نفسه، كما أنه ليس من معالج يستطيع تنفيذ برنامجين أو أكثر بشكل متزامن. أما **التزامن** فيعني «خلال الفترة الزمنية نفسها». إن في استطاعة بعض الطلاب أن يدرسوا ويشاهدوا التلفزيون بصورة متواقتة، ومن المؤكد أن المعالج قادر على تنفيذ برنامجين أو أكثر بصورة متواقتة أيضاً.

أنظمة التشغيل ذات البرمجة المتعددة

إن فوائد البرمجة المتعددة واضحة للغاية: إذ أنه يمكن تنفيذ المزيد من البرامج خلال المدة الزمنية نفسها على الكمبيوتر نفسه. غير أن موارد الكمبيوتر محدودة بالرغم من وفرتها، لذلك لا مفرّ من التزاحم على وقت المعالج، والفسحة المتاحة في الذاكرة الرئيسية، وتخصيص الأجهزة المحيطية عند وجود اثنين أو أكثر من المستخدمين المتواقتين. فينبغي حلّ مشاكل التزاحم هذه عند حدوثها، وبما أنه ليس في استطاعة مشغّلي الكمبيوتر العمل بسرعات عمل الكمبيوتر، فإنه يعود للكمبيوتر نفسه اتخاذ القرارات الرئيسية، ولأن **نظام تشغيل** الكمبيوتر يقوم بوظيفة بينية كيان مادي/كيان منطقي فإن موقعه «الوسطي» (البيني) يجعله المكان النموذجي لتنفيذ **إدارة الموارد**.

الشكل 10.3 يحتوي معظم أنظمة التشغيل البدائية على معالج أوامر ونظام دخل/خرج. وتقوم أنظمة التشغيل ذات البرمجة المتعددة على هذا الأساس.

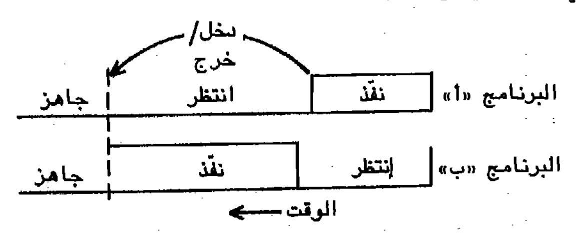


لقد صادفنا أنظمة التشغيل للمرة الأولى في الفصل السادس حيث درسنا وحدتين: معالج الأوامر ونظام التحكم بالدخل/الخرج (الشكل 10.3). إن معالج الأوامر ونظام التحكم بالدخل / الخرج في كمبيوتر رئيسي أكثر تعقيداً من تلك التي تجدها في ميكروكمبيوتر لأن الكمبيوتر الرئيسي مركب من عدد كبير جدا من المكونات (قارن على سبيل المثال بين لغة آيبيإم للتحكم بالمهام ولغة الأمر في نظام التشغيل القرصي الذي أنتجته شركة ميكروسوفت (MS/DOS)، لكن الوظائف التي يؤديانها متطابقة. فنظام التشغيل ذو البرمجة المتعددة مبني على هذا الأساس، ويضيف وحدات لادارة وقت المعالج والفسحة المتاحة في الذاكرة الرئيسية والأجهزة المحيطية. لنبدأ بوقت المعالج.

إدارة وقت المعالج

إفترض أن البرنامجين «أ» و«ب» موجودان في الذاكرة وأن البرنامج «أ» طلب قبل بعض الوقت معطيات من القرص (الشكل 10.4) وكلّفت قناة بعملية الدخل وانتقل المعالج إلى البرنامج «ب». لنفترض أن عملية الدخل قد أنجزت للتو وأن البرنامجين جاهزان للتنفيذ، فأي منهما يحصل على المعالج؟

الشكل 10.4 مع تعدّد المستخدمين المتزامنين من المحتمل ان يكون برنامجان جاهزين للتنفيذ في الوقت نفسه، وعندما يحدث ذلك، ينبغي أن تحل وحدة نظام تشغيل مشكلة التزاحم وأن تحدّد أي البرنامجين ينفذ أولاً.



الشكل 10.5 يُدار وقت المعالج بوحدة نظام تشغيلي.

معالج اوامر	نظام تحكم بالدخل/الخرج	إدارة المعالج

نظام التشغيل

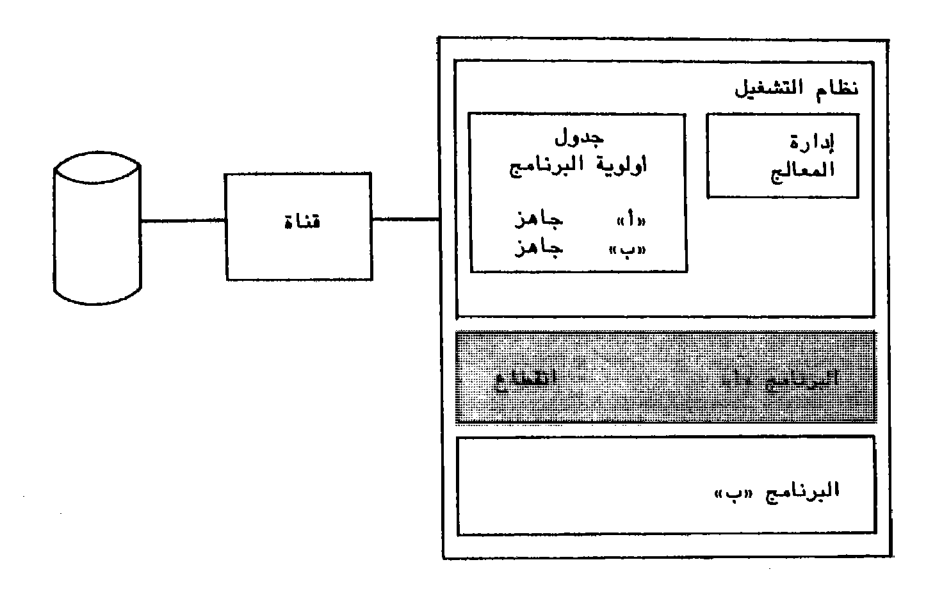
إن أحد الحلول المحتملة هو عرض رسالة على الكونسول تطلب إلى المشغل اتخاذ قرار، وسيحتاج المشغل الى ثانية أو ثانيتين على الأقل للاستجابة، وفي أثناء تلك الثانية أو الثانيتين يمكن أن يكون المعالج قد نفّذ تعليمات البرنامجين معاً بكل سهولة، فالكمبيوترات تبلغ من السرعة درجة يصعب معها على البشر الانتقاء بفاعلية ضمن فترة زمنية حقيقية. وتدير وقت المعالج وحدة نظام تشغيل على معظم الأنظمة ذات البرمجة المتعددة (الشكل 10.5).

كيف يعرف نظام التشغيل متى يجب الانتقال من برنامج الى آخر؟ إن السرّ الكامن وراء ذلك هو الدخل/الخرج. فالبرنامج يفقد التحكم على المعالج بالفعل عندما يبدأ عملية دخل/خرج ومن المحتم أن يستعيد التحكم عندما تنتهي تلك العملية. وإذا كان على نظام التشغيل أن يحدّد البرنامج الذي سينفّذ لاحقاً فينبغي أن يعرف متى تبدأ عمليات الدخل أو الخرج ومتى تنتهي. وتستخدم إشارات إلكترونية تُسمّى **انقطاعات** لكي تميّز هذه العمليات، وعندما يستشعر بانقطاع ما ينقل الكيان المادي التحكم أوتوماتياً إلى نظام التشغيل مهما يكن العمل الذي يعمل الكمبيوتر على إنجازه.

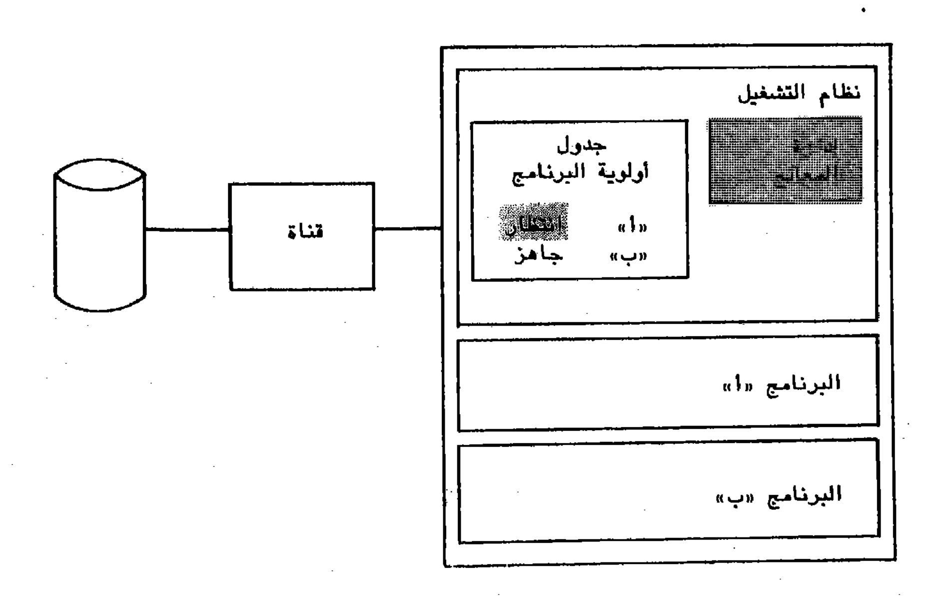
تتبع الخطوات في الشكل 10.6، فعندما يحتاج برنامج تطبيقي للمعطيات يصدر انقطاعاً (الشكل أ 10.6)، وينقل الكيان المنطقي التحكم الى نظام التشغيل استجابة للانقطاع. وما أن يتولى نظام التشغيل التحكم حتى يدخل البرنامج التطبيقي في **حالة انتظار** (الشكل ب 10.6)، ويبدأ عملية الدخل أو الخرج، ويسلم التحكم إلى برنامج تطبيقي آخر (الشكل ج 10.6) ولاحقاً، عندما تنتهي عملية الدخل/الخرج، تصدر القناة انقطاعاً (الشكل د 10.6)، ومرة أخرى يتولى نظام التشغيل التحكم (الشكل 10.6 هـ) ويعطي البرنامج الذي يحتاج المعطيات علامة «جاهز»، ومن ثم ينقل التحكم إلى برنامج تطبيقي (الشكل و 10.6).

ويستخدم نظام التشغيل خوارزمية ليقرّر البرنامج التالي. على سبيل المثال، إذا كانت البرامج جميعها مدرجة في جدول في ترتيب أولوي، فإن نظام التشغيل يستطيع أن يمسح هذا الجدول بعد كل انقطاع ويختار أول برنامج «جاهز» يصادفه. ويجب أن تكون قواعد الأولوية قائمة على معايير معينة مثل حجم البرنامج أو موقعه في الذاكرة أو وقته في الذاكرة أو أهمية المعلومات التي يعطيها. إن النقطة الرئيسية في هذه المسألة هي أن نظام التشغيل يتولى اتخاذ قرار الأولوية بسرعة الكمبيوتر.

الشكل 10.6 إن مفتاح إدارة وقت المعالج هو في معرفة متى تبدأ عمليات الدخل والخرج ومتى تنتهي. ويُشار الى هذه العمليات الحاسمة عموماً بإشارات انقطاع.

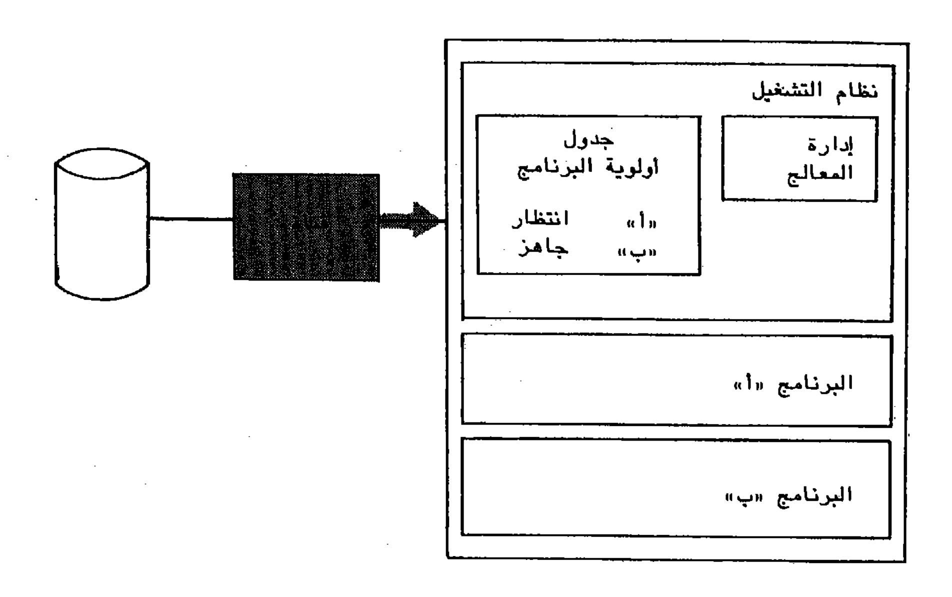


أ. يصدر البرنامج التطبيقي انقطاعاً عندما يحتاج للمعطيات.

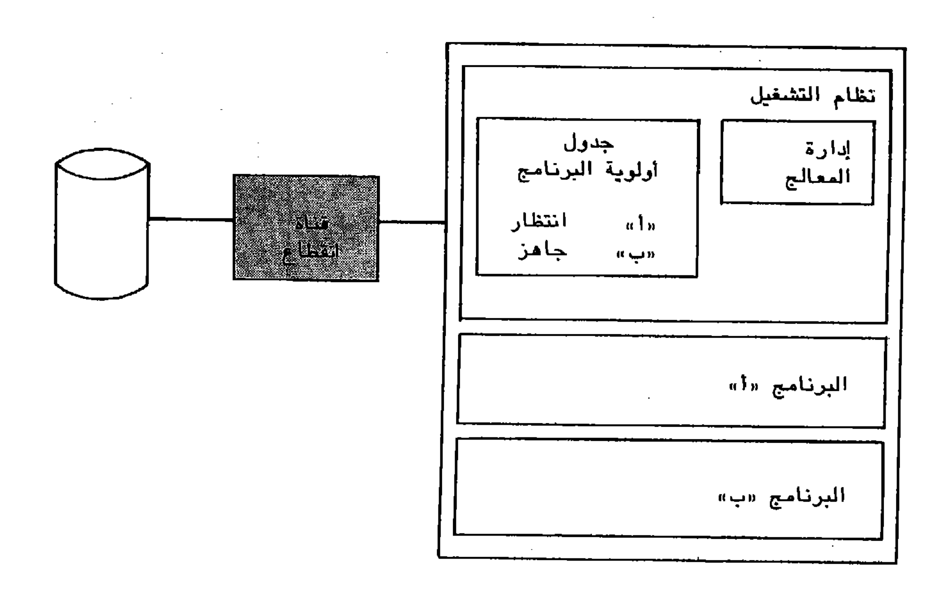


ب. في أعقاب الانقطاع، يتولى نظام التشغيل التحكم ويدخل البرنامج في حالة انتظار.

الشكل 10.6

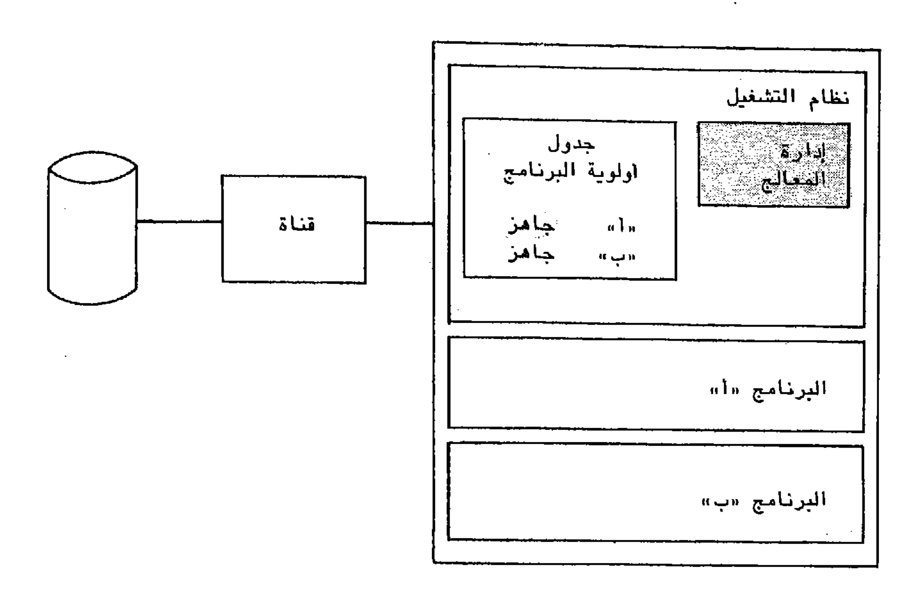


ج. بعد بدء عملية الدخل أو الخرج المطلوبة يسلم نظام التشغيل التحكم إلى برنامج تطبيقي آخر.

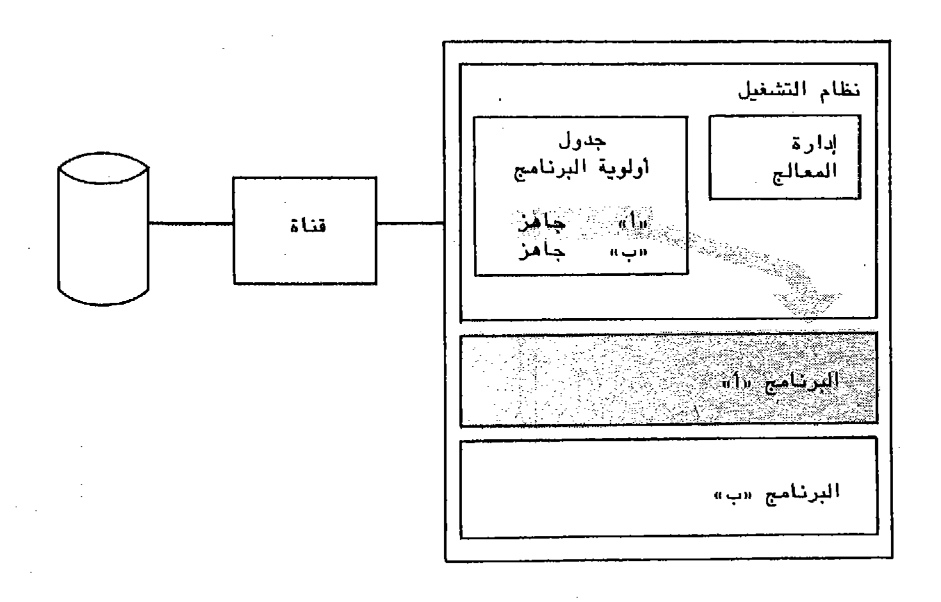


د. تشير القناة من وقت الى آخر إلى نهاية عملية الدخل/الخرج بإرسال انقطاع إلى الكمبيوتر.

الشكل 10.6



هـ في اعقاب الانقطاع يتولى نظام التشغيل التحكم ويعيد البرنامج «ا» إلى حالة «جاهز».

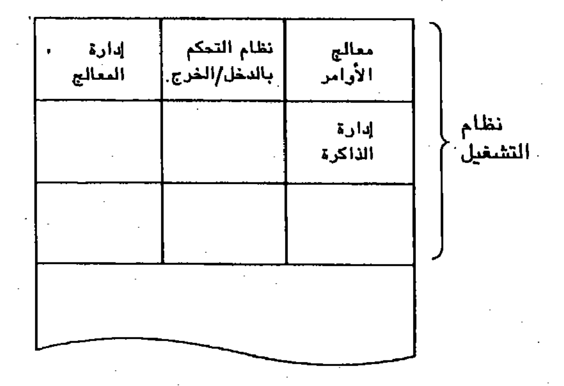


و في النهاية يختار نظام التشغيل برنامجاً تطبيقياً ويسلّمه التحكم بالمعالج.

إدارة الذاكرة

تهتم **إدارة الذاكرة** بتخصيص الفسحة المتاحة في الذاكرة الرئيسية للبرامج التطبيقية، وهي من مسؤولية نظام التشغيل كما هي الحال مع **إدارة المعالج** (الشكل 10.7). وتقسم الطريقة الأبسط، وهي **إدارة الذاكرة بالتقسيم الثابت** (الشكل 10.8) الفسحة المتاحة إلى أقسام ذات أطوال ثابتة، وتخزن برنامجاً واحداً في كل قسم. ويتم التوصل الى ادارة ذاكرة أكثر كفاية مع **إدارة الذاكرة الدينامية**، وباستخدام هذا الأسلوب تعامل الذاكرة المتاحة وكأنها حوض كبير من الفسحات الحرة، ويخصص لكل برنامج ما يحتاجه منها بالضبط.

الشكل 10.7 إدارة الذاكرة هي مسؤولية أخرى من مسؤوليات نظام التشغيل.

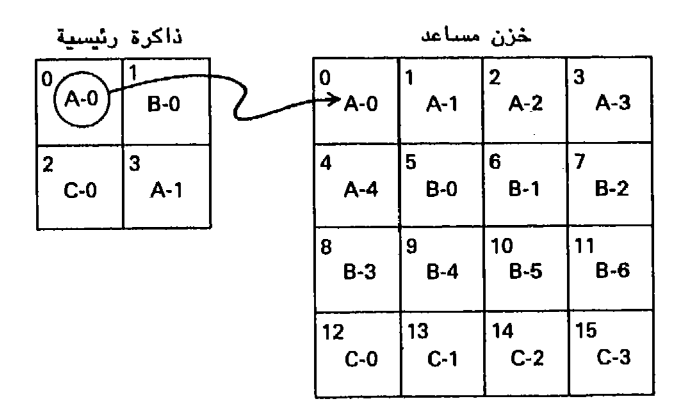


الشكل 10.8 في ظل إدارة الذاكرة بالتقسيم الثابت، تقسم فسحة الذاكرة الرئيسية المتاحة إلى سلسلة من الأقسام ذات الأطوال الثابتة.

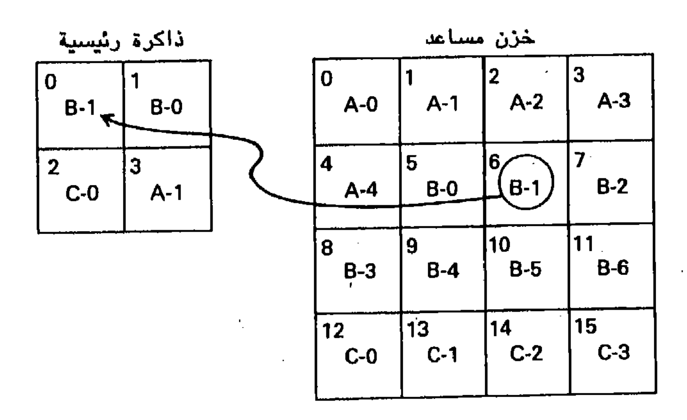


إذا كان الكمبيوتر لا يستطيع أن ينفذ إلا تعليمة واحدة في آن معين، فلماذا ينبغي أن يحمل البرنامج بكامله في الذاكرة الرئيسية؟ ولماذا لا تحمل الأجزاء الفاعلة منه فقط؟ في أنظمة **الذاكرة الظاهرية**، توضع البرامج التطبيقية في خزن مساعد وتنقل أجزاء منها إلى الذاكرة الرئيسية وفقاً للحاجة (الشكل 10.9). وتتطلب هذه البرامج «الجزئية» فسحة في الذاكرة أقل من الذي تتطلبه البرامج الكاملة، وبما أن الفسحة المطلوبة لكل برنامج أقل، يمكن تحميل المزيد من البرامج في الذاكرة الرئيسية، ووجود عدد أكبر من البرامج يعني استعمالاً للمعالج أكثر كفاية.

الشكل 10.9 في نظام ذاكرة ظاهرية، تنتقل أجزاء البرامج ذهاباً وإياباً بين الذاكرة والخزن المساعد.



أ. ينسخ الجزء غير المطلوب من البرنامج إلى خزن مساعد...



ب. ... ويحل مكانه في الذاكرة الرئيسية جزء مطلوب من برنامج آخر.

الشكل 10.10 يخصّص نظام التشغيل أيضاً أجهزة الدخل والخرج والخزن المساعد وينظم البرامج في صفوف وجداول لتحميلها في الذاكرة الرئيسية، كما يؤدي عدداً من وظائف المساندة الأخرى.

تخصيص أجهزة الدخل/الخرج

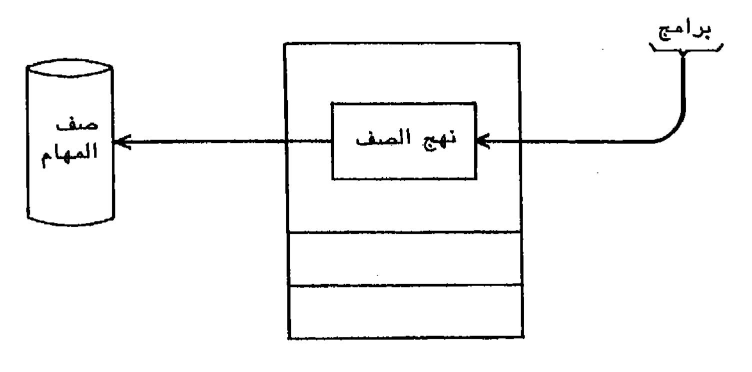
ما الذي كان سيحدث فيما لو تناوب برنامجان على الطابعة نفسها؟ عندئذ سيكون الخرج عديم الجدوى. أو مثلاً تخيل البرنامجين «أ» و«ب» يتناوبان على قراءة وكتابة المعطيات من الشريط نفسه وإليه. فلا شك أن النتيجة ستكون فوضى. ولا يمكن السماح بحدوث ذلك، إذ يجب أن يُدار نيل أجهزة الدخل/الخرج بعناية، ومرة أخرى يكون نظام التشغيل مسؤولاً (الشكل 10.10).

الجدولة

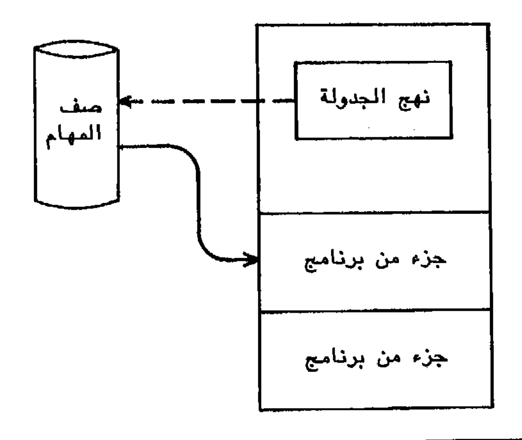
تهتمّ إدارة المعالج بالأولويات الداخلية للبرامج الموجودة حالياً في الذاكرة الرئيسية. فأي برنامج يحمل لاحقاً في الذاكرة بعد أن تنتهي معالجة برنامج ويصبح الحيز متاحاً؟ إن هذا القرار يتعلق عادة بوحدتين مستقلتين هما نهج **صف المهام** ونهج **الجدولة** (الشكل 10.10).

مع دخول البرامج إلى النظام، يقوم نهج صف المهام بترتيبها في صف (الشكل 10.11)، وعندما تكون الفسحة متيسرة، يقوم المجدول باختيار برنامج من الصف ويحمله الى الذاكرة (الشكل 10.12). وغالباً ما يكون البرنامج الأول في الصف هو أول برنامج يحمل، وتستعمل قواعد أولوية أكثر تطوراً في أغلب الأحيان لتحديد البرنامج الذي يحمل لاحقاً. ولا يعود المجدول مسؤولاً عن البرنامج حالما يتم تحميل هذا الأخير في الذاكرة. فبدلاً من ذلك يحدد مدير المعالج في نظام التشغيل حق البرنامج في نيل المعالج.

الشكل 10.11 عندما يدخل برنامج للمرة الأولى في نظام ذي برمجة متعددة، يقوم نهج الصف بنسخه في صف على جهاز خزن مساعد.



الشكل 10.12 عندما تصبح الفسحة متيسّرة في وقت لاحق، يقوم نهج الجدولة بتحميل برنامج من الصف الى الذاكرة الرئيسية.



الخزن المؤقت السريع

إن فائدة البرمجة المتعددة هي أنه يصبح من الممكن تنفيذ عدد أكبر من البرامج ضمن الفترة الزمنية نفسها. وإذا كان في الامكان زيادة سرعة الانتقال بين هذه البرامج، فإنه في الامكان أيضاً بلوغ مستويات كفاية أعلى. تخيل على سبيل المثال نظاماً معيناً ينفذ خمسة برامج متزامنة وكل برنامج يشغل الذاكرة لعشر ثوان، فما أن ينتهي تنفيذ برنامج حتى يحل مكانه برنامج آخر في الذاكرة، وبالتالي يستطيع الكمبيوتر تنفيذ ثلاثين برنامجاً في الدقيقة، وإذا ما استطعنا تقليص وقت تنفيذ كل برنامج الى خمس ثوان، فانه يصبح باستطاعتنا تنفيذ ستين برنامجاً في تلك الدقيقة نفسها.

تصور برنامجاً يولد كشف رواتب لـ1000 موظف. إن قراءة 1000 بطاقة ساعات عمل تستغرق دقيقتين على الأقل، وتستغرق طباعة 1000 شك بضع دقائق اضافية، وبالتالي سيحتاج البرنامج لأربع أو خمس دقائق لكي يتنفذ. ولكن ماذا لو استبدلت قارئة البطاقات والطابعة البطيئتان بقرص؟ إن مدوار الأقراص أسرع بكثير، وهكذا يتنفّذ البرنامج في وقت أقل بكثير، وبالنتيجة تتحرّر الذاكرة للعمل على برنامج آخر بسرعة أكبر.

هذه هي الفكرة الرئيسية وراء عملية **الخزن الموقت السريع**. فمن الشائع حتى مع البرمجة المتعددة أن تكون البرامج التطبيقية في انتظار الدخل/الخرج. وعند حدوث ذلك، لا يكون للمعالج عمل يقوم به. وفي أثناء هذه الفترات تقرأ وحدة الخزن الموقت السريع في نظام التشغيل المعطيات من الأجهزة البطيئة كقارئات البطاقات أو كلوحات مفاتيح الطرفيات وتخزنها على وسيط فائق السرعة كالقرص، حتى قبل أن يكون البرنامج الذي يحتاج هذه المعطيات قد حمل في الذاكرة. وفي وقت لاحق يمكن قراءة معطيات دخل البرنامج من القرص عندما يحمل في الذاكرة، وعند الخرج تنقل المعطيات بالخزن الموقت السريع إلى قرص ومن ثم تفرغ بعد التشغيل الى الطابعة. ولأنه لا يتعامل الا مع الدخل/الخرج الفائق السرعة فقط، فإن البرنامج التطبيقي ينتهي من المعالجة بسرعة أكبر بكثير ويخلي الفسحة لبرنامج آخر.

المشاركة الزمنية

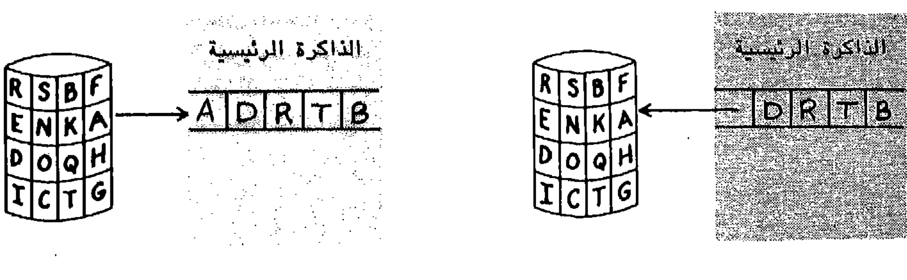
ربما تكون قد استخدمت طرفية لكتابة برامج أصلية أو تنفيذ كيان منطقي موجود أو نيل معطيات، ومن الموكد أن طرفيتك، بالاضافة الى العشرات أو ربما المئات غيرها من الطرفيات كانت موصولة بكمبيوتر مركزي. إن مثل هذه الترتيبات تتطلب عادة **مشاركة زمنية**.

تخيّل تطبيقاً نموذجياً للمشاركة الزمنية. تتم طباعة العمليات - عبارات البرامج المستقلة أو سطور معطيات الدخل أو الأوامر - عبر لوحة مفاتيح، وفي معظم الحالات يتطلب الأمر قدراً قليلاً فقط من المعالجة الفعلية. ان الطباعة وأكثر ما يستطيع البشر فعله هو طباعة عمليتين في الدقيقة، أما بالنسبة للكمبيوتر فيمثل كل مستخدم سلسلة من طلبات المعالجة الوجيزة المتباعدة. وفي أثناء معالجة عملية يعرف النظام أن وقتاً طويلاً سيمضي قبل تلقي عملية المستخدم التالية، وهكذا يمكن نقل فسحة العمل من الذاكرة الى الخزن المساعد، مما يفسح في المجال لتطبيق آخر في الذاكرة الرئيسية (الشكل أ 10.13)، وفي وقت لاحق يعاد نقل حيز عمل المستخدم الى الذاكرة الرئيسية ويسلم التحكم اليه عندما تصل عملية المستخدم التالية (الشكل ب 10.13)، وتستخدم أساليب **النقل الى الذاكرة والنقل من الذاكرة** هذه على معظم الأنظمة التي تعمل بالمشاركة الزمنية لادارة فسحة الذاكرة الرئيسية.

وتخيّل انك أمضيت لتوك عشرين دقيقة في طباعة المعطيات لبرنامج تحليل احصائي، وان كل عنصر معطيات كان عملية موجزة فقط، فمن الموكد ان عملك حتى هذه النقطة يتناسب وافتراضات عمل مشاركة زمنية نموذجي. غير أن عمليتك الاخيرة مختلفة، فهي أمر يأمر النظام بأن يعالج المعطيات، وهو يدفع الكمبيوتر الى الشروع في نهج حسابي يمكن أن ينفذ لدقيقتين أو ثلاث دقائق، وأثناء معالجة عمليتك، سيكون على كل من المستخدمين الآخرين للنظام أن ينتظر، وهذا ما لا يمكن تحمّله.

إن الحل هو في **توزيع الحصص الزمنية**، ويقتصر كل برنامج على «حصة» قصوى من الزمن، ربما 0.01 من الثانية، وإذا أنجزت المعالجة خلال هذه الفترة فذلك أمر حسن، ويجري انتقال التحكم الى برنامج آخر. غير انه اذا لم تنجز المعالجة فان البرنامج يقاطع وينقل الى آخر الصف لينتظر دوراً آخر. ان توزيع الحصص الزمنية هو تقنية ادارة المعالج المستخدم على الأنظمة التي تعمل بالمشاركة الزمنية.

الشكل 10.13 النقل الى الذاكرة والنقل من الذاكرة. عندما ينتهي النظام من معالجة عملية، تنقل عمل المستخدم من فسحة الذاكرة الى قرص (أ) وفي وقت لاحق تنقل العمل مجدداً الى فسحة الذاكرة عندما تدخل عملية المستخدم التالية الى النظام.



أ. نقل من الذاكرة.

ب. نقل الى الذاكرة.

المعالجة المتعددة

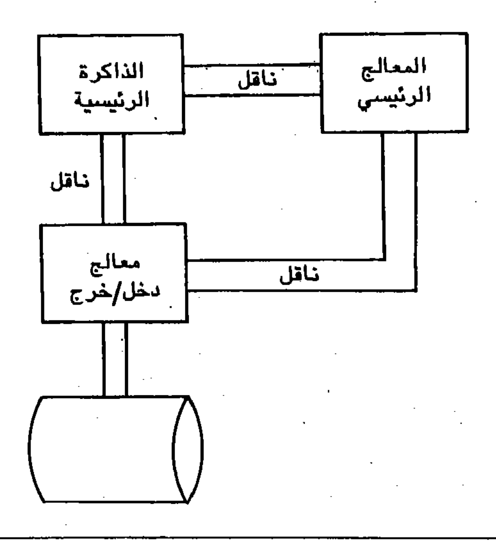
تتطلب البرمجة المتعددة والمشاركة الزمنية نظام تشغيل مقيم للتعامل مع الصراعات التي تظهر عندما يتقاسم الموارد المحدودة عدد من المستخدمين المتزامنين. يساعد الأسلوبان على تحسين كفاية النظام، ويسمحان بمعالجة المزيد من البرامج خلال الفترة الزمنية نفسها وعلى الكيان المادي نفسه. غير أن وحدات نظام التشغيل تحتل الذاكرة الرئيسية وتستهلك وقت المعالج بالرغم من كونها أساسية، وهي تمثل تكاليف ثابتة غير ذات مردود.

لنأخذ على سبيل المثال مسألة التحكم بالدخل/الخرج. لقد طوّرت القنوات لتحرير المعالج من الكثير من هذه المسؤولية، ولكن لسوء الحظ لا تستطيع القناة القيام بالمهمة كلها لوحدها، وحتى الآونة الاخيرة كان المعالج الرئيسي الذي يعمل تحت تحكم نظام التشغيل يؤدي وظائف منطقية معينة مثل البدء والانهاء والتحقق من وضع عملية الدخل/الخرج. فلماذا لا نحدّد كود نظام التشغيل الذي يؤدي هذه الوظائف، ونبرمج معالجاً ميكروياً للقيام بالأمور نفسها؟ علينا أن نستبدل معالج الدخل/الخرج الجديد هذا بالقناة (الشكل 10.14). والآن وبعد أن أصبح لدينا معالجان مستقلان يمكن تنفيذ التعليمات المرتبطة بالدخل/الخرج بموازاة نشاطات المعالج الرئيسي الأكثر انتاجية. ويتقاسم المعالجان الذاكرة الرئيسية نفسها فيشكلان نظاماً ذا **معالجة متعددة**.

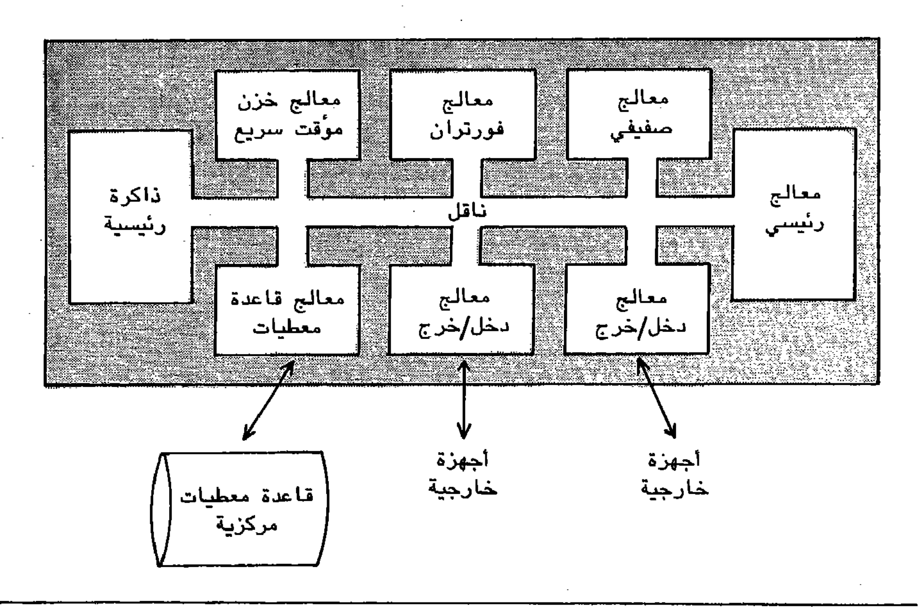
إذا كان في الامكان نقل التحكم بالدخل/الخرج الى معالج مستقل، فلماذا لا تنقل وظائف أخرى (الشكل 10.15)؟ قد تتضمن أنظمة المستقبل القريب عدة معالجات دخل/خرج، ليحل واحد منها محل كل قناة. وقد يتحكم معالج قاعدة المعطيات، الذي يجسد معظم منطق ادارتها، بكل الاتصالات مع قاعدة المعطيات. وفي المسائل العلمية والهندسية الضخمة، يمكن لمعالج صفيفي أن يعتق المعالج الرئيسي من مهمة معالجة الصفيفيات الروتينية المستهلكة للوقت، ويمكن للمعالجات اللغوية أن تسمح بتنفيذ البرامج المكتوبة بلغة عالية المستوى مباشرة، متجاوزة بذلك عملية التصريف غير المجدية.

واذا كان كثير من الوظائف المرتبطة الآن بأنظمة التشغيل وبرامجيات النظام تنقل الى معالجات مستقلة، فان برامج التحكم تلك (برامج التكاليف الثابتة) لن تحتجز أبداً بعدها وقت المعالج الرئيسي، وستكون النتيجة قدراً أكبر من الكفاية.

الشكل 10.14 يمكن للمعالج الرئيسي أن يتحرّر تماماً من المسؤولية عن التحكم بعمليات الدخل/الخرج اذا ما استبدل معالج دخل/خرج بقناة.



الشكل 10.15 في المستقبل القريب سيوكل الكثير من الوظائف الثابتة الى معالجات مستقلة وبذلك يتحرّر المعالج الرئيسي للعمل على البرامج التطبيقية.



وهناك مناظرة مشوقة في ما يتعلق بالوجهة المستقبلية لحقل الكمبيوتر - هل تكون السيطرة للكمبيوترات الصغيرة أم الكبيرة؟ والجواب هو أن هناك احتمالاً أن تكون للاثنين معاً. ان الأنظمة ذات البرمجة المتعددة كبيرة بحكم تعريفها، ومع ذلك فان المكونات الرئيسية، والمعالجات الثانوية، هي في الحقيقة ميكروكمبيوترات، والواقع ان آلات صغيرة تستخدم لتركيب آلات كبيرة. وفي الفصل التالي سوف ندرس الشبكات ونكتشف تزاوجاً مشوقاً آخر بين تقنيات الكمبيوترات الصغيرة والكبيرة.

الخلاصة

ان الكمبيوتر الرئيسي أسرع بكثير من أجهزته المحيطية، وهو بالنتيجة يمضي وقتاً أطول في انتظار الدخل/الخرج منه في معالجة المعطيات. ان أحد الحلول المتوافرة هو البرمجة المتعددة، ويحمل برنامجان أو أكثر في الذاكرة الرئيسية وينفذهما المعالج بصورة متزامنة، محولاً انتباهه من برنامج الى آخر. وبسبب تنافس برامج متعددة على موارد الكمبيوتر المحدودة، فانه لا مفر من الصراعات التي يتم حل معظمها بواسطة نظام التشغيل.

ان إحدى وظائف نظام التشغيل الهامة هي ادارة وقت المعالج. فعندما تبدأ عملية دخل أو خرج يتم اشعار نظام التشغيل بواسطة انقطاع، وعند تسلّمها التحكم تضع البرنامج الذي يطلب الدخل/الخرج في حالة انتظار وتسلم التحكم بالمعالج الى برنامج آخر، وفي نهاية الأمر يُشار الى نهاية عملية دخل/خرج بانقطاع آخر. ومرة ثانية يتسلم نظام التشغيل التحكم ويشير الى البرنامج المؤاتي بعلامة «جاهز» ويسلم التحكم مجدداً الى برنامج تطبيقي. ثم تختار وحدة ادارة المعالج البرنامج «التالي» باستخدام خوارزمية أولويات.

وتقسم ادارة الذاكرة في أبسط اشكالها الذاكرة الرئيسية الى أقسام ثابتة الطول ويحمل برنامج واحد في كل منها. ويمكن تحقيق كفاية أكبر باستخدام ادارة الذاكرة الدينامية. وفي ادارة الذاكرة الظاهرية تخزّن البرامج على قرص وتحمل الأجزاء الفاعلة فقط في الذاكرة الرئيسية، وتخصّص وحدة نظام تشغيل أخرى الأجهزة المحيطية للبرامج التطبيقية.

يجري خزن البرامج على صف عندما تدخل نظاماً ذا برمجة متعددة، وفيما بعد، وعندما تصبح فسحة الذاكرة متاحة تقوم وحدة مجدول باختيار البرنامج التالي من الصف وتحمله في الذاكرة الرئيسية. ولتحسين سرعة الانتقال بين البرامج، تنقل المعطيات في أغلب الأحيان من أجهزة بطيئة السرعة الى قرص، ومن ثم تقرأ من القرص الى البرنامج، وفي جانب الخرج، تنقل النتائج الى قرص وتفرغ عادة الى الطابعة.

وتستخدم المشاركة الزمنية غالباً في الأنظمة المتعددة الطرفيات، ويمكن ادارة فسحة الذاكرة باستخدام أسلوبي النقل من الذاكرة والنقل الى الذاكرة لأن الوقت بين العمليات المتلاحقة عادة طويل جداً. ويستخدم توزيع الحصص الزمنية في ادارة وقت المعالج للتخلص نهائياً من خطر احتكار برنامج واحد للنظام وإكراه جميع المستخدمين الآخرين على الانتظار.

وفي ظل المعالجة المتعددة يتقاسم معالجان أو اكثر ذاكرة مشتركة، ومن الممكن أن تقلص المعالج المتعددة الى حد بعيد خسارة وقت المعالج في الوظائف الثابتة غير المنتجة.

مصطلحات أساسية

- ☐ متزامن
- ☐ ادارة الذاكرة الدينامية
- ☐ ادارة الذاكرة بالتقسيم الثابت
- ☐ انقطاع
- ☐ تخصيص جهاز الدخل/الخرج
- ☐ ادارة الذاكرة
- ☐ المعالجة المتعددة
- ☐ البرمجة المتعددة
- ☐ نظام التشغيل
- ☐ ادارة المعالج
- ☐ الصف
- ☐ ادارة الموارد
- ☐ نقل من الذاكرة/ نقل الى الذاكرة
- ☐ الجدولة
- ☐ الخزن المؤقت السريع
- ☐ المشاركة الزمنية
- ☐ توزيع الحصص الزمنية
- ☐ ذاكرة ظاهرية
- ☐ حالة انتظار

اختبار ذاتي

1. على أية مهمة يمضي كمبيوتر نموذجي معظم وقته؟

أ. تنفيذ التعليمات ج. تفسير التعليمات
ب. التصريف د. انتظار الدخل/الخرج

2. في ________، يحمل برنامجان أو أكثر في الذاكرة الرئيسية وينفذان بصورة متزامنة.

أ. المعالجة المتعددة ج. التصريف
ب. البرمجة المتعددة د. تعدد المهام

3. تُدار موارد الكمبيوتر بواسطة ________.

أ. مبرمجيه ج. كيانه المنطقي
ب. كيانه المادي د. نظام تشغيله

4. يخصّص وقت المعالج بواسطة ________.

أ. الكيان المادي ج. الكيان المنطقي
ب. المشغل د. نظام التشغيل

5. يمكن أن يتحول نظام التشغيل من برنامج الى برنامج عن طريق الاستجابة لـ________

أ. برنامج ج. المشغل
ب. كيان مادي د. انقطاعات

6. أبسط انواع ادارة الذاكرة هي ادارة الذاكرة ________

أ. المبنية على الكتل ج. المبنية على القطاعات
ب. بالتقسيم الثابت د. الدينامية

7. في ظل ادارة الذاكرة ________، يخصّص لكل برنامج ما يحتاجه من الذاكرة.

أ. الديناميةج. الظاهرية
ب. الثابتة د. بأسلوب النقل من/النقل الى

8. في إدارة الذاكرة ________، تخزّن البرامج على قرص وتحمل الأجزاء الفاعلة فقط في الذاكرة.

أ. الدينامية ج. الظاهرية
ب. بالتقسيم الثابت د. كل الأشكال المذكورة

9. عندما تدخل البرامج نظاماً ذا برمجة متعددة يتم خزنها على ________

أ. شريط ج. صف
ب. سكة د. لا شيء من هذه

10. تحمل البرامج في الذاكرة الرئيسية بواسطة ________ في نظام التشغيل.

أ. نهج الصف ج. مدير المعالج
ب. مدير الذاكرة د. المجدول

11. تنسخ المعطيات من أجهزة بطيئة السرعة الى أجهزة فائقة السرعة بواسطة ________ لتستعمل كدخل لبرنامج في وقت لاحق.

أ. الجدولة ج. المشاركة الزمنية
ب. الصف د. الخزن الموقت السريع

12. يحتمل أن يكون النظام الذي يضم مائة طرفية أو اكثر ويشغله كمبيوتر مركزي نظام ________

أ. برمجة متعددة ج. معالجة متعددة
ب. مشاركة زمنية د. توزيع حصص زمنية

13. يستخدم معظم أنظمة المشاركة الزمنية ________ لادارة الذاكرة.

أ. التقسيمات الثابتة ج. النقل من/النقل الى الذاكرة
ب. المناطق الدينامية د. الصفحات

14. ان أسلوب ادارة المعالج المستخدم على معظم أنظمة المشاركة الزمنية هو أسلوب ________.

أ. توزيع الحصص الزمنية ج. معالجة الأوامر
ب. معالجة الانقطاعات د. لا شيء مما سبق

15. في ــــــــــــــــ، يتقاسم معالجان مستقلان أو أكثر الذاكرة نفسها.

أ. المعالجة المتعددة
ب. البرمجة المتعددة
ج. الحساب المتعدد
د. تعدد المهام

الاجابات

1.د 2.ب 3.د 4.د 5.د 6.ب 7.أ 8.ج 9.ج 10.د 11.د 12.ب 13.ج 14.أ 15.أ.

ربط المفاهيم

1. تخيل أن ذاكرة كمبيوتر رئيسية تحتوي برنامجاً واحداً فقط. إن المعالج لا يستطيع القيام بشيء في أثناء عمليات الدخل والخرج. لماذا؟
2. اشرح البرمجة المتعددة بايجاز.
3. فرّق بين المصطلحين «متواقت» و«متزامن»، ولماذا يعد هذا الفرق مهماً؟
4. لماذا يعتبر نظام التشغيل مكاناً جيداً لتطبيق ادارة الموارد؟
5. ما هو الانقطاع؟ ولماذا تعتبر الانقطاعات مهمة لادارة المعالج على نظام متعدد البرمجة؟
6. يخزّن الجزء الأكبر من برنامج معين على جهاز مساعد في نظام الذاكرة الظاهرية، وفي الواقع لا تحمل في الذاكرة الرئيسية إلّا الأجزاء الفاعلة. لماذا يعد ذلك مهماً؟
7. فرّق بين الجدولة والخزن الموقّت السريع.
8. تهتم ادارة المعالج بأولوية البرامج الموجودة في الذاكرة الرئيسية في حين تهتم الجدولة والصف بأولوية البرامج غير الموجودة بعد في الذاكرة الرئيسية. اشرح.
9. اشرح المشاركة الزمنية باختصار. كيف تدار فسحة الذاكرة في نظام مشاركة زمنية؟ وكيف يُدار وقت المعالج؟
10. فرّق بين البرمجة المتعددة والمعالجة المتعددة.

11.

الأنظمة الموزّعة

مفاهيم أساسية

الاتصالات المعطياتية

عملية وصل الطرفيات وأجهزة الكمبيوتر

الاتصالات المعطياتية لكيان منطقي

الشبكات

الاتصالات المعطياتية

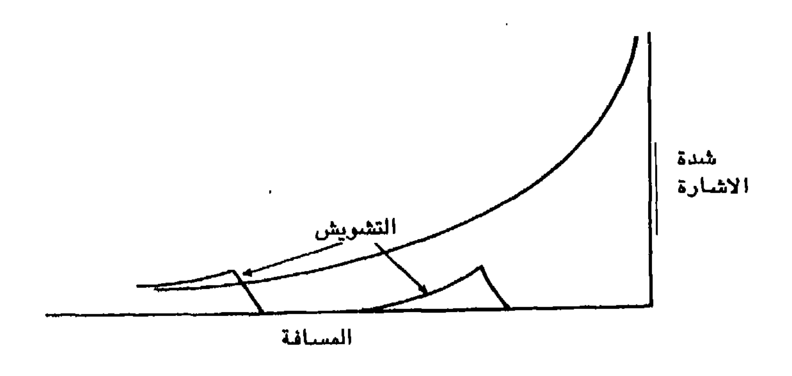
في نظام مشاركة زمنية، ينال المستخدمون كمبيوتر ما عبر الطرفيات. وتكون بعض الطرفيات **محلية**، موصولة مباشرة بكبلات الى الكمبيوتر، بينما تكون طرفيات أخرى **بعيدة**، فتتصل مع الكمبيوترات البعيدة عبر خطوط التلفون أو أي وسيلة ارسال أخرى. ويتضمن العمل بالكمبيوتر عن بعد عدداً من المشاكل. فلنتحرى عنها.

تتمثل المعطيات، ضمن كمبيوتر، بنبضات متقطعة الكترونية - 0s و1s. وبما ان مكونات النظام تبعد عادة شيئاً قليلاً عن بعضها البعض، يسهل حينها نقل تلك الخوينات من مكون الى آخر. وعلى اي حال، فنحن عندما نحاول ارسال نبضات الكترونية عبر مسافة ما، تحدث عدة أشياء. أولاً، تفقد الاشارة الشدة أو تخف بسبب مقاومة السلك (الشكل 11.1). وفي نفس الوقت، تلتقط التداخل أو **التشويش**. ان الشواش الذي يحدث في خلفية محطة راديو بعيدة هو مثال جيد على ذلك. ويزداد ضعف الاشارة كلما بعدت عن مصدرها، ويصبح التشويش أكثر حدة الى أن تصبح الاشارة في آخر الأمر مطغى عليها. إذا ما أريد ارسال المعطيات عبر مسافة ما، يجب ترشيح التشويش وتعزيز الاشارة ذاتياً من حين الى آخر.

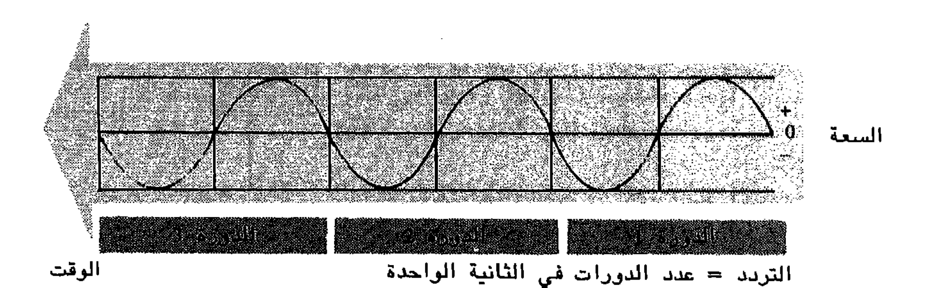
ترسل المعطيات غالباً في سياق **إشارة الموجة الحاملة** مثل الموجة الجيبية في الشكل 11.2. يسمى نموذج «S على طرفها» **بالدورة**. ويكون علو الموجة من النقطة الأكثر انخفاضاً الى النقطة الأكثر ارتفاعاً هو سعتها. ويشكل عدد الدورات في الثانية تردد الموجة. وبما ان اشارة الحامل يتم ارسالها بتردد وسعة محددين فانه من الممكن تصميم جهاز يعمل على ترشيحها وتعزيزها ذاتياً.

كيف يمكن استخدام خصائص الموجة هذه لتكويد وارسال المعطيات الثنائية؟ ان الأمر بغاية البساطة. ابدأ مستخدماً موجة مقياسية لها سعة وتردد محددين ودع كل دورة تمثل خوينة أحادية. ولارسال خوينة 1، اترك الموجة لمفردها ولارسال خوينة 0، نوع تردد دورة واحدة (الشكل 11.3). وبمعنى آخر، فان الدورة العادية تمثل خوينة 1 وشيئاً آخراً، ألا وهي خوينة 0. وتكون النتيجة اشارة مستمرة تمثل نمطاً من الخوينات.

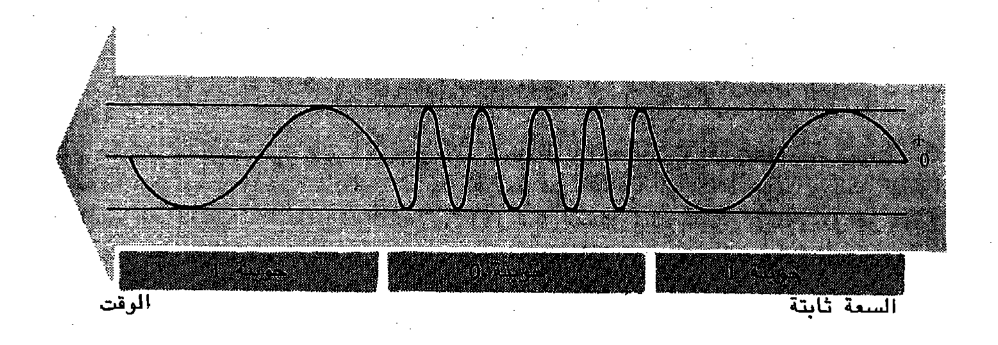
الشكل 11.1 تميل اشارة الكترونية تمرّ في سلك الى فقدان الشدة او انخفاضها بسبب مقاومة السلك. ويطغى التشويش في آخر الأمر على الاشارة ليصبح من غير الممكن ارسال المعطيات.



الشكل 11.2 يتم ارسال المعطيات غالباً في محيط اشارة حاملة مثل الموجة الجيبية هذه.



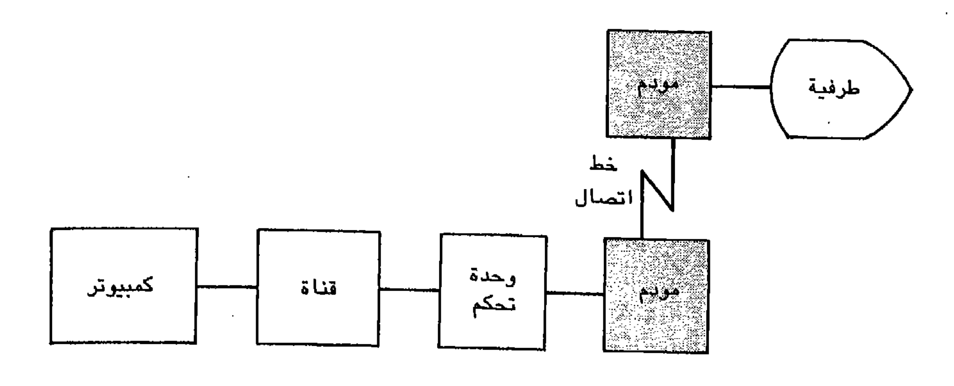
الشكل 11.3 يمكن ارسال المعطيات الثنائية بتنويع عمود اشارة الحاملة اختيارياً.



إن المعطيات هي نبضات الكترونية متقطعة. والموجة هي **نظيري** يمثل المعطيات. ونحن نستعمل النظيريات كل يوم. فارتفاع عمود من الزئبق في ميزان حرارة ليس الحرارة الفعلية بل انه يمثلها. كما إن موقع الابرة في لوحة التحكم بالسيارة ليس هو السرعة بل يمثلها. ان موجة متواصلة تمرّ عبر خط اتصال لا تمثل المعطيات ولكنها تكون مشابهة لها.

وعلى أي حال، فان الكمبيوترات لا تخزّن المعطيات كموجات متواصلة، بل انها تخزن نبضات متفرّدة وتعالجها. وكلما تم ارسال المعطيات ما بين كمبيوتر وطرفية بعيدة فانه يجب بسبب التضارب الالكتروني هذا تحويل هذه المعطيات من شكل النبضة الى شكل الموجة والعكس بالعكس. يُسمى التحويل الى شكل الموجة **بالتضمين**؛ أما التحويل الى شكل النبضة فيسمى **بازالة التضمين**. ويسمى جهاز الكيان المادي الذي يقوم بتنفيذ هذه المهمة **بمجموعة المعطيات** أو **بالمودم** (مضمن/مزيل التضمين). ويوجد عادة مودم واحد في كل طرف من خط الاتصال (الشكل 11.4).

الشكل 11.4 يستخدم المودم لتضمين اشارة او ازالة التضمين منها. ويكون هناك عادة مودم على كل طرف من خط الاتصال.

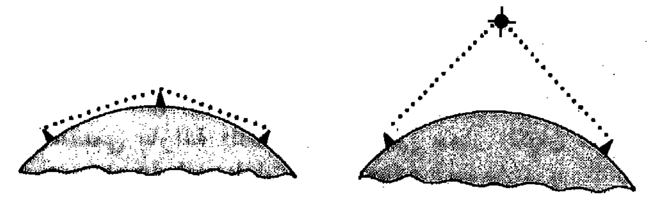


يرسل الكثير من خطوط الاتصال اشارات نظيرية لأنه حتى مؤخراً لم يكن باستطاعتنا بناء معدات بشكل اقتصادي تتعامل مع الاشارات **الرقمية**. ان التقدم في ميدان الالكترونيات لم يجعل ارسال المعطيات الرقمية ممكناً فحسب، بل وشائعاً. وفي الواقع، فان معظم مرافق الاتصالات الحديثة هي رقمية. وبوجود الاتصال الرقمي، تزول موجة حامل متواصلة. وبدلاً من ذلك فانه يتم ارسال الخوينات كنبضات مختصرة متقطعة وتكون هذه الاشارات الرقمية أقل تأثراً بالتشويش، لذا فانه يتم ارسال المعطيات بكثير من الدقة.

ان شبكة الهاتف هي على الأرجح أكثر وسائل **الاتصالات المعطياتية** شهرة. يقدر خط طبقة صوتية نموذجية بـ2400 خوينة تقريباً في الثانية أو 2400 **بود**. ويمكن للقنوات ذات السرعة العالية والنطاق الواسع، من الارسال بمعدل يبلغ مليون بود، وتتوافر معدلات أبواد عدة ما بين هذين الحدّين. ان ارسال المعطيات بواسطة الموجات الميكروية هي وسيلة بديلة عن خطوط التلفون. لكن ولسوء الحظ، فان الارسال بواسطة الموجات الميكروية هو ارسال محصور بخط منظور. وكما نعلم جميعاً، فان الأرض مستديرة وفيها انحناء. هذا الانحناء يقيد من مدى الموجات الميكروية، فيجعل صنع محطات الترحيل المكلفة أو أقمار الاتصال الصناعية أمراً ضروريا (الشكل 11.5).

يمكننا أن نقضي وقتاً طويلاً في شرح وسائل اتصال متنوعة، لكن هذا الاجراء سيكون مربكاً دون موجب. بدلاً من ذلك، فاننا سنستعمل مصطلحاً عاماً وهو «**الخط**»، لوصف أي وسيلة اتصال معطيات.

الشكل 11.5 ان إرسال المعطيات بواسطة الموجات الميكروية ارسالاً محصوراً بخط منظور. يتطلب ارسال الموجات الميكروية من مسافات بعيدة محطات ترحيل او اقمار صناعية.



عملية وصل الطرفيات وأجهزة الكمبيوتر

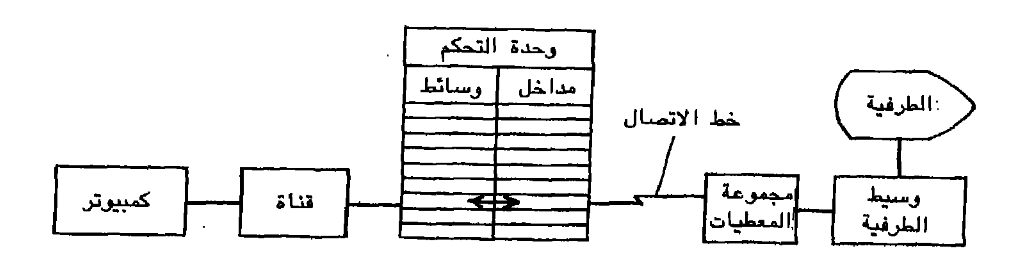
تخيّل عدة طرفيات وقد وصلت الى كمبيوتر مركزي. يمكن معالجة المعطيات في طرف الكمبيوتر بسرعات تبلغ ملايين الرموز في الثانية. كما ان خطا نموذجيا من 2400 بود يمكنه ارسال 300 رمز تقريباً في الثانية. وماذا عن الطرفية؟ لنفترض ان معظم المستخدمين يطبعون بسرعة فان ارسال 10 رموز في الثانية هو أمر مبالغ فيه على الأرجح. لدينا لوحة مفاتيح ذات 10 رموز في الثانية موصولة بخط ذي 300 رمز في الثانية والذي بدوره موصل الى كمبيوتر ذي ملايين من الرموز في الثانية. انه حقاً لمدى مدهش.

ان **وسيطاً** ذا حجم يكفي لاحتواء معاملة أو شاشة ممتلئة بالمعطيات يمكن أن يساعد في جعل الطرفية وخط اتصالها متزامنان (الشكل 11.6). وبينما يطبع المستخدم، تحفظ الرموز في وسيط. وبعد انتهاء عملية الطباعة، يقوم المستخدم بالضغط على مفتاح الادخال. ورداً على هذه الاشارة، ترسل محتويات الوسيط عبر الخط بالسرعة المقدرة له. وعند وضع الخرج، تنتقل المعطيات عبر الخط وتدخل الوسيط. ويتم عرضها من هناك أو طباعتها على حسب السرعة المقدرة للطرفية. وبوجود وسيط في الوسط، يزول التفاوت في السرعة ما بين الطرفية والخط.

ماذا يحدث على الطرف الآخر من الخط؟ توجد هناك **وحدة تحكم بالارسال**، تُسمّى عادة **بجهاز الطرف الأمامي**، تتركب من سلسلة من المداخل والوسائط المترابطة (الشكل 11.6). ان **المدخل** هو نقطة وصل في خط اتصال، يحتوي عادة على الالكترونيات التي يحتاج اليها لتضمين ولازالة التضمين عن اشارة ما. تدخل المعطيات الى وحدة التحكم بالارسال بمعدل يحدّد خط الاتصال وتنتقل الى الوسيط المترابط مع مدخل المبدأة.

وعندما تصبح كل المعطيات موجودة في الوسيط، ترسل وحدة التحكم بالارسال اشارة الى القناة التي بدورها تقوم بارسال اشارة الى الكمبيوتر تفيد بأن المعطيات أصبحت جاهزة للادخال. وحالما يصبح الكمبيوتر جاهزاً، يتم ارسال المعطيات عبر خط ناقل بسرعة المعالجة الداخلية للكمبيوتر. في وضع الخرج، تنتقل المعطيات الى وحدة التحكم بسرعة الكمبيوتر، وتوزع على خطوط الاتصال بسرعات أقل بكثير.

الشكل 11.6 يساعد وسيط لطرفية على جعل سرعتَي الطرفية والخط متزامنين. وتوجد على طرف الكمبيوتر وحدة تحكم بالارسال تحتوي على وسيط لكل مدخل.



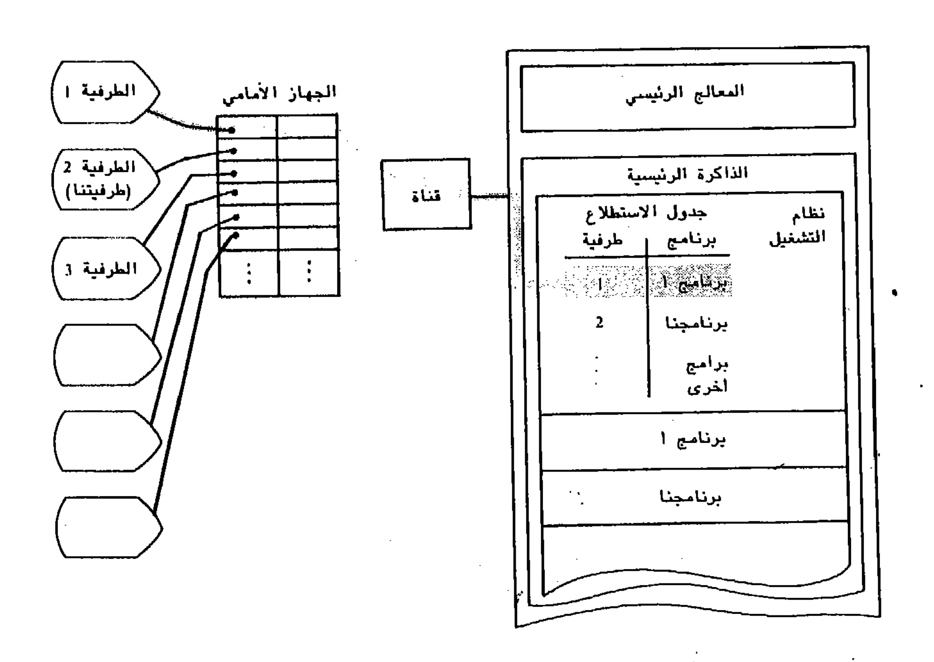
الاتصالات المعطياتية لكيان منطقي

يتضمّن نظام مشاركة زمنية نموذجي عدة طرفيات وخطوط اتصال موصولة الى كمبيوتر بواسطة جهاز أمامي واحد أو أكثر. ان مجرد التنسيق بين كل الكيانات المادية هذه هو أمر صعب. وبالاضافة الى ذلك، فان كل طرفية قد تساند مستخدم مستقل يقوم بنيل برنامج مستقل. ان المعطيات المخصّصة للبرنامج A لا تصلح للبرنامج B، فالدخل والخرج يجب توجيههما الى برامج معينة. وغالباً ما يتحكم نظام التشغيل بارتباط معطيات الكيان المادي بالكيان المنطقي من خلال عملية تُعرف **بالاستطلاع**.

إبدأ بطرفية. بينما يجري طبع المعطيات تدخل الى وسيط الطرفية. وأخيراً، وبعد طبع الرمز الاخير، يضغط المستخدم على مفتاح الادخال. فيؤدّي هذا الى تشغيل مفتاح الكتروني، وتعطي الطرفية علامة بانها مستعدة.

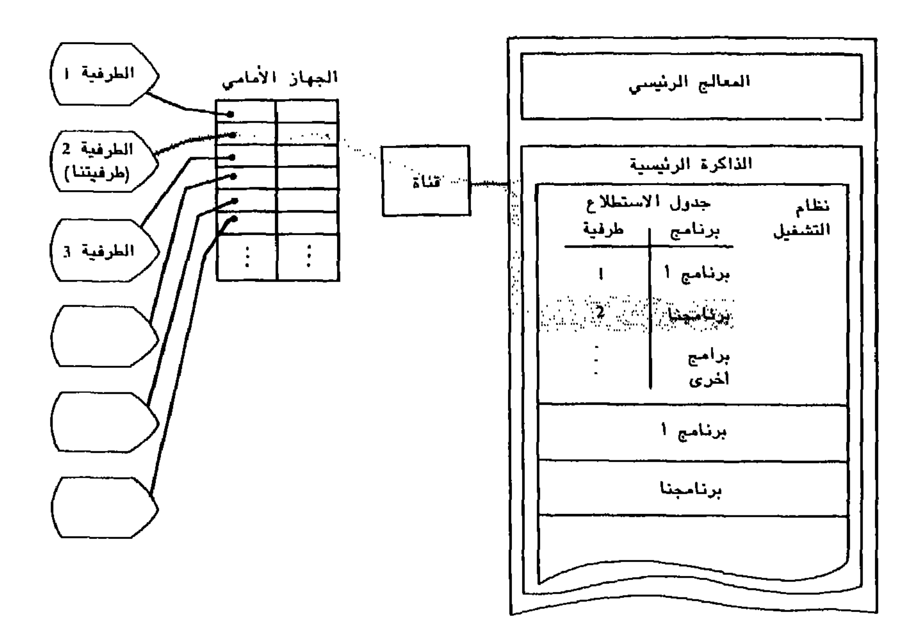
والآن انتقل الى داخل جهاز الكمبيوتر. تجد نظام التشغيل متحكماً بالوضع، ولديه جدول يدرج اسم كل برنامج نشط وطرفيته المرافقة له. وبالاشارة الى هذا الجدول، ترسل اشارة استطلاع الى طرفية البرنامج الأول (الشكل أ 11.7). في الواقع تسأل الاشارة الطرفية اذا ما كانت مستعدة لارسال المعطيات. سنفترض ان المستخدم لا زال يطبع. وبما ان مفتاح الادخال لم يضغط عليه بعد، فان مفتاح الاستعداد غير مشغل. وهكذا ينتقل نظام التشغيل الى طرفية البرنامج الثانية ويصدر اشارة استطلاع أخرى. هذه الطرفية الثانية هي طرفيتنا، وهي على استعداد (الشكل ب 11.7) وهكذا يتم قبول التبادل ويوجّه الى برنامجنا (الشكل ج 11.7).

الشكل 11.7 يستطلع الكمبيوتر الطرفيات ليحدّد أية طرفية ستقوم بالاتصال بالكمبيوتر التالي.

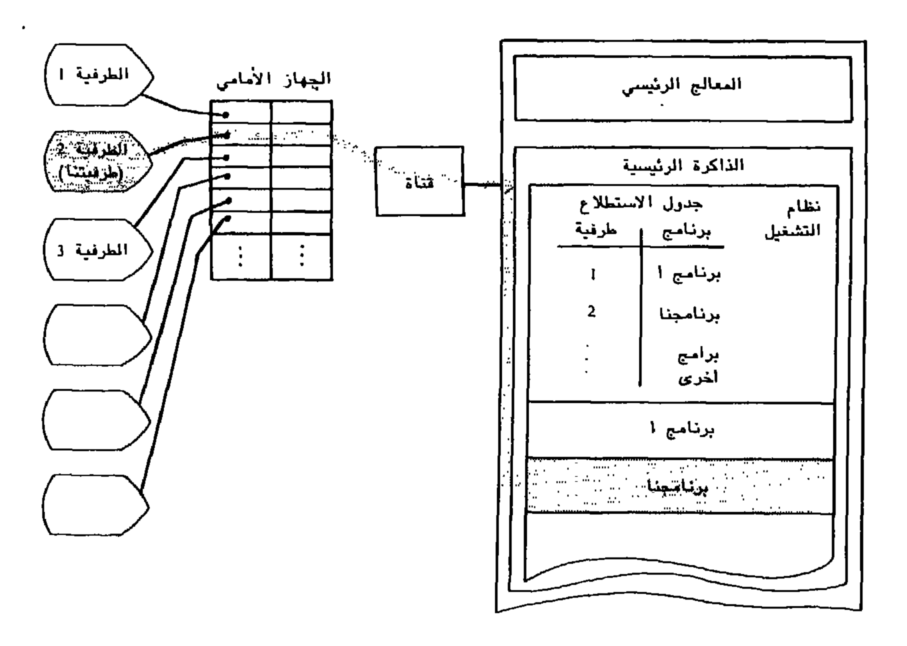


أ. ترسل اشارة استطلاع الى الطرفية التي تساند البرنامج الأول في جدول الاستطلاع.

الشكل 11.7



ب. وبما ان الطرفية الأولى غير مستعدة، ترسل اشارات استطلاع الى طرفية البرنامج الثانية.



ج. وبما ان هذه الطرفية مستعدة، تتم قراءة معطياتها للبرنامج.

لا يستطيع أن يفعل الكمبيوتر شيئاً بدون برنامج ما وبدون توليد اشارات استطلاع ايضاً. ويكون نهج الاستطلاع عادة جزءاً من نظام التشغيل؛ إلا ان توليد نظام التشغيل للاشارات يؤدي الى هدر الكثير من وقت المعالج الرئيسي الثمين. وكحل اختياري يعطي الجهاز الأمامي معالجاً مستقلاً خاصاً به وتكليفه بمهمة الاستطلاع. وفي أنظمة كهذه، يتصل المعالج الأمامي مباشرة مع الطرفيات ويستطلعها باستمرار وبالتعاقب. اذا كانت الطرفية غير مستعدة، تهمل الى أن تبدأ دورة الاستطلاع التالية؛ واذا كانت مستعدة، تنقل معطياتها الى وسيط المعالج الأمامي.

وفي داخل الكمبيوتر، يجب أن يقرّر نظام التشغيل (أو وحدة كيانات منطقية أخرى تُدعى «**برنامج مراقبة الاتصالات المعطياتية**») أي البرامج سيحصل على التحكم في الدورة التالية. تذكر، ان كل برنامج مرتبط بمدخل أمامي خاص. فاذا لم تصل بعد معطيات برنامج ما الى وسيطها الأمامي، لن يكون هناك من داع لاعطاء امكانية التحكم لذلك البرنامج. وهكذا يرسل مراقب الاتصالات المعطياتية اشارة الاستطلاع الخاصة به الى المعالج الأمامي ليسأله في الواقع ما اذا كانت المعطيات موجودة في وسيط لمدخل ما. فاذا وصلت المعطيات، تصبح هي الدخل ويسيطر برنامج التطبيق المناسب على التحكم. وإن لم تصل، يتم استطلاع المدخل التالي للجهاز الأمامي. لاحظ ان الكمبيوتر الرئيسي يتصل مع المعالج الأمامي فقط، وبسرعة كمبيوترية دائمة. فالكمبيوتر عليه أن لا ينتظر طرفية ما أو خط اتصال ما على الاطلاق. وبدلاً من ذلك، على المعالج الأمامي، الأقل كلفة، أن ينتظر الاستجابة من مكونات نظام أشد بطءاً.

الشبكات

كانت أجهزة الكمبيوتر في الماضي غالية الثمن الى حد ان معظم المؤسسات الكبرى كانت تنفذ كل معالجات معطياتها في جهاز مركزي واحد. وبينما كانت طريقة الجهاز المركزي فعالة جداً فيما يخص جدول الرواتب وتوليد تقارير حسابية فانها لم تكن ذات فائدة لأولئك الذين هم بحاجة الى استجابة سريعة لبرنامج فريد محلي. وبوجود الميكروكمبيوترات والمينيكمبيوترات المنخفضة الثمن، وفي الوقت الحاضر، لم يعد هناك من سبب يحول دون اقتنائها بواسطة مكتب فرعي أو دائرة هندسة أو أي مؤسسة أخرى تكون بحاجة لمساعدة الكمبيوتر. بربط هذه الآلات البعيدة الى كمبيوتر مركزي عبر خطوط الاتصال، يمكن مراقبة النشاطات المحلية وتنسيقها. تُسمى هذه الطريقة **بمعالجة المعطيات الموزعة**.

علينا أن نلاحظ ان عملية معالجة المعطيات الموزعة تتطلب عدداً من الكمبيوترات الموصولة بخطوط اتصال. كما ان بوجه عام عدة كمبيوترات مستقلة موصولة بخطوط اتصال تشكّل **شبكة**.

قد تكون أفضل طريقة لتخيّل شبكة هي في اعتماد مثال ما. لنستخدم نظام فحص في سوبرماركت يتحكم به الكمبيوتر. تكون كل محطة فحص (الشكل 11.8) مجهزة بماسح كود قضيبي. وتبدأ عملية الفحص حين يلتقط الموظف منتجاً ما، ثم يحدد كود الانتاج العالمي الخاص بها (UPC)، ويمرّره فوق الماسح الذي يقوم بقراءة الكود.

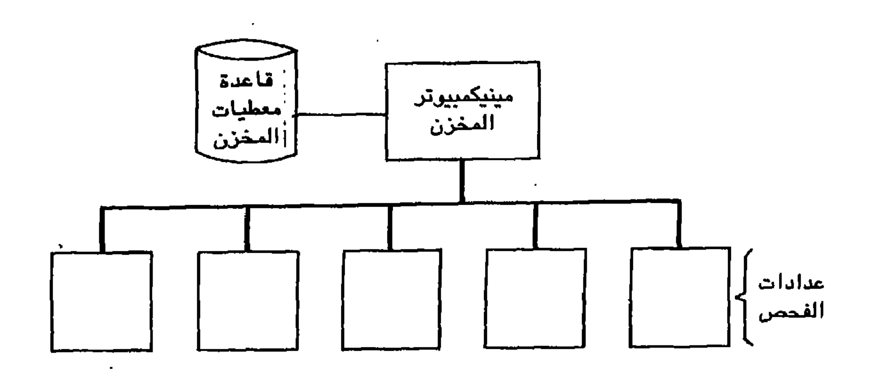
تكون محطات الفحص موصولة الى مينيكمبيوتر يتحكم بالنيل لمجمع معطيات يحتوي، على سعر المبيع الحالي وعلى الوصف وعلى كمية الموجودات وعلى ضريبة المبيعات وعلى غيرها من المعطيات عن كل صنف في المخزن حسب كود الانتاج العالمي (الشكل 11.9). وفي حين يمسح كود منتج ما، يقوم الدخل في المينيكمبيوتر، الذي يحدد المنتج في قاعدة المعطيات، باضافة السعر وضريبة المبيعات على فاتورة الزبون وبطرح 1 من كمية موجودات المنتج. والخطوة التالية تكون في اعادة الوصف والسعر والضريبة

الى محطة الفحص، حيث يتم طبع وصل الزبون. وتستمر هذه الدورة الى أن يتم مسح الترتيب العام. وبما ان عملية المسح يتحكم بها الانسان، فهي بطيئة نسبياً، لذلك يكون من السهل على المينيكمبيوتر ملاحقة العديد من محطات الفحص.

الشكل 11.8 ان محطات الفحص في السوبرماركت كالتي تراها في الشكل، تقوم بمسح كود الانتاج العالمي المطبوع على معظم البضائع، وبذلك تؤمن معطيات الدخل الأساسية لشبكة السوبرماركت.



الشكل 11.9 عندما يتم مسح كود الانتاج العالمي، يتوجه الكود الى مينيكمبيوتر المخزن الذي يقوم بالبحث عن سعر المنتج وعن الوصف وعن أي معطيات أخرى في قاعدة المعطيات.

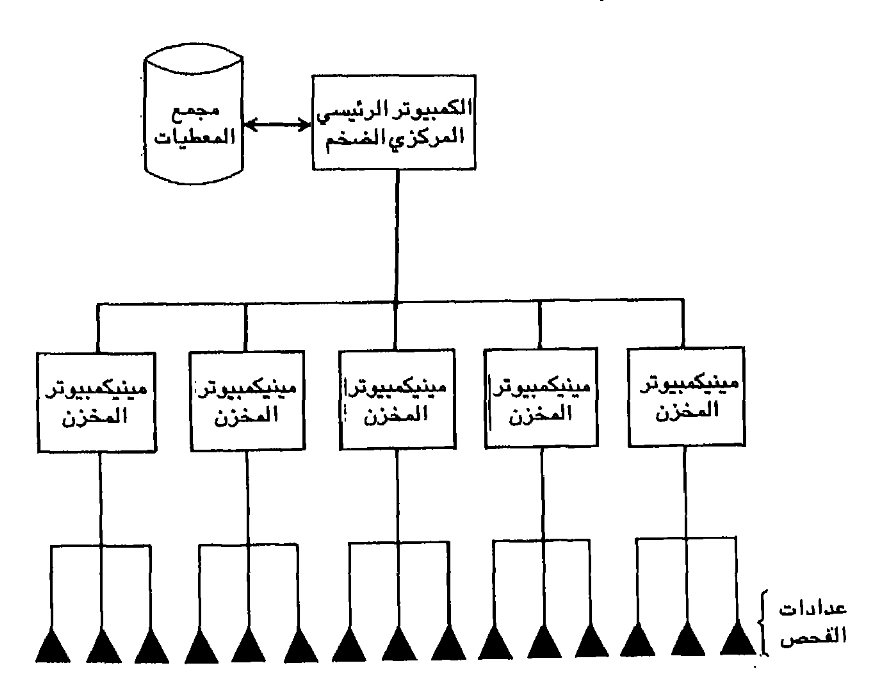


مع ان نظام فحص ما في سوبرماركت يتحكم به كمبيوتر مفيد للمخزن، فان قيمته الفعلية تكون في تحسين ضبط خزن البضائع والتوزيع. وعموماً يكون الشراء بكميات أقل كلفة من شراء صنف واحد في كل مرة. لذا تقوم فروع السوبرماركت المتعددة بشراء المنتجات بشكل شاحنات صندوقية وتخزّنها في المستودع ثم توزعها على المخازن المتفرقة وتبيعها مع ربح معين. قد يكون تخزين المنتجات وتوزيعها مكلفاً، وعلى أي حال، فان كل درهم يصرف على التخزين أو الشحن أو التعبئة يكون درهم ربح محتمل ضائع. ان فروع السوبرماركت الناجحة تبقي كلفة التخزين والتوزيع في أدنى حد ممكن.

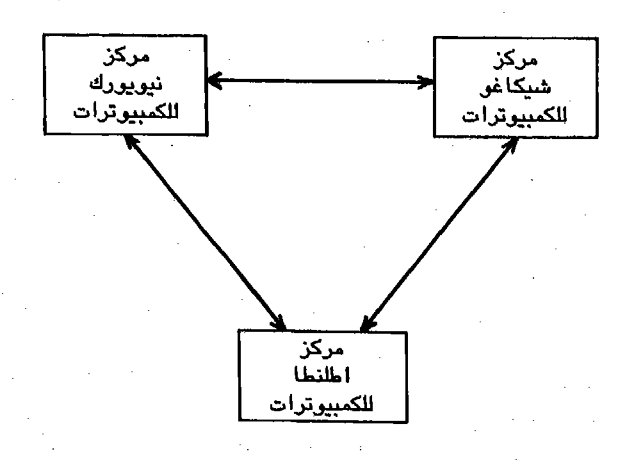
وتكون المشكلة بغاية البساطة بوجود بضعة منتجات فقط وعدد قليل من المخازن. على أي حال، خذ في الاعتبار عدد المنتجات الموجودة في سوبرماركت نموذجي واضربها بعدد فروع المخازن، وحاول الاحتفاظ بالمعلومات عن كل المنتجات في كل هذه المخازن. ان أساليب حل مشاكل معقدة كهذه تتطلب كمبيوتراً رئيسياً.

وهكذا تلجأ الى اضافة عامل ثالث الى شبكة السوبرماركت، الا وهو الكمبيوتر الرئيسي المركزي (الشكل 11.10). ويكون من ثم مسؤولاً عن التحكم بالموجودات في المستودع المركزي وفي المخازن كلها. كما انه ينظّم عمليات التوزيع الى المخازن. وترسل المعطيات التي تساند هذه التطبيقات الى الكمبيوتر الرئيسي من مينيكمبيوترات المخازن عبر خطوط الاتصال.

الشكل 11.10 يوجد على رأس شبكة السوبرماركت كمبيوتر رئيسي ضخم يتعامل مع لائحة الموجودات وبعمليات التوزيع.



الشكل 11.11 توصَل عادة الكمبيوترات ذي القوة المماثلة ببعضها البعض لتشكّل شبكة حلقية. وتسمح خطوط الاتصال للكمبيوترات بالاتصال فيما بينها. وتؤمن الشبكة احتياطياً فعالاً اذا ما تعطل كمبيوتر ما.



تتألف شبكة ما من عدة كمبيوترات موصولة بخطوط اتصال. ويمكن للآلات أن تعمل باستقلالية ولكنه بالامكان ايجاد تنسيق بين وظائفها. تجمع محطات الفحص المعطيات في شبكة سوبرماركت فتنفذ المينيكمبيوترات معالجة بمستوى المخزن ويهتم الكمبيوتر الرئيسي الضخم بمهمات الموجودات وتوزيعها. وفي شبكة موزعة أخرى، قد يكون لدى دوائر المحاسبة ودوائر المبيعات ودوائر الانتاج ودوائر الهندسة في شركة أعمال كمبيوترات خاصة بها، وتكون كلها موصولة الى كمبيوتر رئيسي مركزي يحتوي على قاعدة معطيات المؤسسة. وبوجود هكذا نظام، يصبح لدى الدوائر المستقلة المرونة بالقيام بالمعالجات بذاتها، بينما تضمن الآلة المركزية استعمال الدوائر للمعطيات العامة وتساعد في المراقبة والتحكم بنشاطاتها.

وعلى أي حال، فان الشبكات غير مقيدة بهذه التركيبات الهرمية. على سبيل المثال، توصَل عدة مؤسسات كبرى كمبيوتراتها الموزعة في أنحاء العالم بشكل شبكة حلقية (الشكل 11.11). وبوصل كل الكمبيوترات تصبح الأقسام المختلفة، وبشكل واضح لنا، قادرة على تبادل المعلومات. وهناك ما هو غير واضح يكمن في الاحتياطي الذي تؤمنه شبكة كهذه في حال تعطل كمبيوتر ما وذلك بنقل وظيفته الى الكمبيوترات الأخرى.

ليست كل الشبكات خاصة. على سبيل المثال، فان كل شخص لديه كمبيوتر ومودم وكيان منطقي مناسب وتلفون يمكنه الترابط مع أي عدد من الكمبيوترات ونيل معطيات متنوعة تتراوح ما بين تسعير المخزون وفهرس المكتبة الى لوحات نشرات الكترونية. وهناك أنظمة معقدة أكثر تسهل حتى التبادل البريدي الالكتروني بين مستخدمين في جميع أنحاء العالم. لقد بدأنا للتو تلمّس الطاقة الكامنة في شبكات الكمبيوتر.

الخلاصة

توصل الكبلات الطرفيات المحلية مباشرة الى نظام كمبيوتر. وتترابط الطرفيات البعيدة مع كمبيوتر عبر خطوط الاتصالات المعطياتية. تخزن المعطيات كنبضات متقطعة داخل الكمبيوتر، ولكنه غالباً ما يجب تحويلها الى شكل نظيري لأجل ارسالها عبر خطوط الاتصالات. يسمى تحويل النبضات الى الشكل الموجي بالتضمين؛ أمّا التحويل من الشكل الموجي الى النبضات فيسمى بازالة التضمين. ويسمى الجهاز الذي ينفذ عمليات التحويل هذه بالمودم. قد تكون أفضل وسيلة اتصالات معطياتية معروفة هي شبكة التلفون. وهناك بديل لها وهو ارسال الموجات الميكروية.

هناك تفاوت ملحوظ بين سرعة الطرفيات وسرعة خطوط الاتصال وسرعة الكمبيوتر. يمكن للوسائط المساعدة على جعل هذه الأجهزة أو الوسائل المتباينة متزامنة. كما ويمكن لكمبيوتر ما أن يترابط مع طرفية واحدة في وقت واحد. ويختار الطرفية التالية بواسطة عملية استطلاع. وغالباً ما تكون خطوط الاتصال متصلة بنظام كمبيوتر بواسطة جهاز أمامي الذي يأخذ على عاتقه مسؤولية استطلاع الطرفيات. يحتوي الجهاز الأمامي على مدخل ووسيط لكل خط. وفي داخل الكمبيوتر، يترابط نظام التشغيل أو مراقب اتصالات معطياتية مع الجهاز الأمامي، ويستطلع المداخل ليحدّد ما اذا كانت معطيات برنامج ما موجودة في الوسيط.

ان الشبكة هي مجموعة كمبيوترات تم وصلها بواسطة خطوط الاتصال. وقد استخدمنا نظام فحص في سوبرماركت لتوضيح شبكة هرمية. وفي شبكة حلقية، يتم وصل عدة كمبيوترات ذي قدرة متساوية تقريباً؛ وهي تؤمن الاتصال والاحتياطي.

مصطلحات أساسية

- ☐ نظيري
- ☐ بود
- ☐ وسيط
- ☐ اشارة حاملة
- ☐ دورة
- ☐ اتصالات معطياتية
- ☐ برنامج مراقبة اتصالات معطياتية
- ☐ مجموعة معطيات
- ☐ إزالة التضمين
- ☐ رقمي
- ☐ معالجة معطيات موزعة
- ☐ جهاز امامي
- ☐ خط اتصال
- ☐ موضعي
- ☐ مودم
- ☐ تضمين
- ☐ شبكة
- ☐ تشويش
- ☐ استطلاع
- ☐ بوابة
- ☐ بعيد
- ☐ وحدة تحكم بالارسال

اختبار ذاتي

1. تترابط طرفية ________ مع كمبيوتر بعيد عبر خطوط الاتصالات المعطياتية.

أ. بعيدة
ب. معروضة
ج. مطبوعة
د. موضعي

2. إن التداخل الذي يشوّه الاشارات الالكترونية المرسلة عبر مسافة ما يسمى ________.

أ. استاتي ج. تشويش
ب. تداخل د. مسمار

3. إن الموجة هي مثال على معطيات ________.

أ. ثنائية ج. رقمية
ب. مرسلة د. نظيرية

4. معظم مرافق الارسالات المعطياتية الجديدة تكون ________.

أ. رقمية ج. موجات ميكروية
ب. نظيرية د. كل ما ورد خطأ

5. في داخل الكمبيوتر تخزّن المعطيات كـ________.

أ. أمواج متواصلة ج. إما أ أو ب
ب. نبضات متقطعة د. كل ما ورد خطأ

6. يحوّل ________ المعطيات من شكل نبضي الى شكل موجي ومن شكل موجي الى شكل نبضي.

أ. مودم ج. قناة
ب. وحدة تحكم د. بينية

7. إن المعيار الأساسي لسرعة الاتصالات المعطياتية هو ________.

أ. خوينات بالثانية ج. رموز بالثانية
ب. خانات بالثانية د. معدل بودي

8. يمكن استخدام المصطلح العام ________ لوصف أية وسيلة اتصالات معطياتية.

أ. شبكة ج. خط
ب. نظيري د. رقمي

9. إن الأجهزة أو الوسائل التي تعمل بسرعات مختلفة يمكن جعلها متزامنة باستخدام ________.

أ. خط اتصال ج. وسيط
ب. مودم د. مجموعة معطيات

10. يمكن وصل عدد من خطوط الاتصال في كمبيوتر ضخم الى نظام أحادي عبر ________.

أ. بينية ج. مينيكمبيوتر
ب. جهاز أمامي د. خط ناقل

11. ________ هو نقط وصل لخط اتصال.

أ. مدخل
ب. وسيط
ج. الطرف الأمامي
د. استطلاع

12. يختار كمبيوتر ما الطرفية التي سيترابط معها بواسطة عملية تُعرف باسم ________.

أ. التنسيق
ب. البينية
ج. إرحال
د. الاستطلاع

13. عدة كمبيوترات يتم وصلها بخطوط اتصال تشكل ________.

أ. شبكة
ب. نظام موزع
ج. نظام مشاركة زمنية
د. حلقة

14. كان نظام الفحص في السوبرماركت، الذي تم شرحه في هذا الفصل، مثالاً على شبكة ________.

أ. حلقية
ب. نجمية
ج. هرمية
د. مشاركة زمنية

15. يؤمّن ترتيب شبكة حلقية ________.

أ. الاتصال
ب. الاحتياطي
ج. أ وب معاً
د. كل ما ورد خطأ

الاجابات

1.أ 2.ج 3.د 4.أ 5.ب 6.أ 7.د 8.ج 9.ج 10.ب 11.أ 12.د 13.أ 14.ج 15.ج.

ربط المفاهيم

1. ميّز ما بين الطرفيات المحلية والطرفيات البعيدة.
2. لماذا هي ضرورية عملية ترشيح وتعزيز الاشارة عند ارسال المعطيات عبر مسافة ما؟
3. ما هي اشارة الحاملة؟ ولماذا تستعمل اشارات الحاملة؟
4. ميّز ما بين المعطيات النظيرية والمعطيات الرقمية.
5. ماذا يفعل المودم؟ وأين تكمن أهمية المودم؟
6. ما هو الوسيط؟ ولماذا يستعمل
7. اشرح باختصار عملية الاستطلاع.
8. ما هو الجهاز الأمامي؟ وما هي الوظائف التي يؤديها؟

9. شرحنا في الفصل 10 فكرة المعالجة المتعددة. تخيل معالج أمامي «شديد الذكاء» يأخذ على عاتقه المسؤولية المترتبة عن عملية الاتصالات المعطياتية. هل هذا الاجراء من قبل معالج أمامي يكون بمثابة معالجة متعددة؟ لماذا أو لما لا؟
10. ما هي الشبكة؟

ملحق

الأنظمة العددية

الأعداد العشرية

يتكوّن العدد العشري من سلسلة من الأرقام ـ 0، 1، 2، 3، 4، 5، 6، 7، 8، 9 ـ مكتوبة في مواضع محدّدة، وهذه المواضع مهمة، فالعددان 23 و32، على سبيل المثال، مختلفان بالرغم من حقيقة احتوائهما الرقمين نفسهما. وتستنتج قيمة عدد معين من خلال ضرب كل رقم بقيمة منزلته أو موضعه وجمع النتائج، فمثلاً يمثل العدد 3582:

٣ ضرب	1000	=	3000	
+ ٥ ضرب	100	=	500	
+ ٨ ضرب	10	=	80	
+ 2 ضرب	1	=	2	
			3582	

وعموماً فان قيمة أي عدد هي بكل بساطة مجموع نتائج ضرب أرقامه وقيمه الموضعية.

أنظر بتمعّن الى القيم الموضعية العشرية 1، 10، 100، 1000، 10000، الخ... ان النمط واضح، فبدلاً من كتابة كل هذه الأصفار نستطيع أن نستعمل الرمز العلمي ككتابة 10000 على شكل 10^4، ولأن أي عدد يرفع للأس صفر هو 1، فنحن نستطيع أن نكتب القيم الموضعية العشرية على شكل الأساس الذي نستخدمه (10) مرفوعاً لسلسلة من الأسس العددية:

$$10^0 \quad 10^1 \quad 10^2 \quad 10^3 \quad 10^4 \quad 10^5 \quad 10^6 \quad 10^7 \quad 10^8 \; \ldots$$

من الممكن اشتقاق عدد قليل من القواعد العامة من خلال بحثنا للنظام العشري. القاعدة الأولى هي فكرة قيمة المنزلة أو القيمة الموضعية ممثلة بالأساس (10) مرفوعاً لسلسلة من الأسس العددية، والقاعدة الثانية هي استخدام الرقم صفر (0) لتمثيل «لا

شيء» في موضع معين (وإلا فكيف كنا سنستطيع أن نميز بين 3 و30)، أما القاعدة الثالثة فهي أن هناك حاجة لما مجموعه عشرة أرقام (0 الى 9) لكتابة القيم العشرية، وفي النهاية ليس من الممكن كتابة أي قيمة برقم واحد ما عدا القيم الأقل من قيمة الأساس فقط (10 في هذه الحالة).

الأعداد الثنائية

ليس هناك من شيء يحصر استخدام هذه القواعد بنظام عددي أساسه 10، فإذا كانت القيم الموضعية أسس للرقم 2، فسيكون لدينا شبكة النظام العددي الثنائي أو الأساس 2:

$$\dots\ 2^8\quad 2^7\quad 2^6\quad 2^5\quad 2^4\quad 2^3\quad 2^2\quad 2^1\quad 2^0$$

وكما هي الحال في أي نظام عددي، هناك حاجة للرقم صفر (0) لتمثيل لا شيء في موضع معين، وبالاضافة الى ذلك يحتاج النظام العددي الثنائي الى رقم آخر واحد فقط هو 1. وبوجود قيم المنازل أو القيم الموضعية هذه، نستطيع أن نجد قيمة أي عدد عن طريق ضرب كل رقم بقيمة منزلته وجمع نتائج الضرب هذه. على سبيل المثال، العدد الثنائي 1100011 هو:

1 ضرب 2^6 = 1 ضرب 64 = 64
+ 1 ضرب 2^5 = 1 ضرب 32 = 32
+ 0 ضرب 2^4 = 0 ضرب 16 = 0
+ 0 ضرب 2^3 = 0 ضرب 8 = 0
+ 0 ضرب 2^2 = 0 ضرب 4 = 0
+ 1 ضرب 2^1 = 1 ضرب 2 = 2
+ 1 ضرب 2^0 = 1 ضرب 1 = 1

99

ان العدد العشري 2 هو 10 في الثنائي، والعدد العشري 4 هو 100 أما العشري 5 فهو 101 (1 أربعة و0 اثنان و1 واحد).

الثماني والست عشري

وبصورة عامة تستخدم في الكمبيوترات أنظمة عددية أخرى أبرزها الثماني (الأساس 8) والست عشري (الأساس 16)، ويستخدم النظام العددي الثماني اسس الرقم 8 لتمثيل القيم الموضعية وقيم الأرقام 0 و1 و2 و3 و4 و5 و6 و7، أمّا النظام العددي الست عشري فيستخدم أسس العدد 16 والأرقام 0 و1 و2 و3 و4 و5 و6 و7 و8 و9 وA وB وC وD وE وF، والعدد الست عشري FF هو:

15 ضرب 16^1 = 240
+ 15 ضرب 16^0 = 15

255

ليس هناك من كمبيوترات تعمل مباشرة بالقيم الثمانية أو الست عشرية، فالكمبيوتر هو آلة ثنائية، ويستخدَم هذان النظامان العدديان بكل بساطة لأنه من السهل التحويل

بينهما وبين الثنائي. وكل رقم ثماني يوازي ثلاثة أرقام ثنائية بدقة، كما أن كل رقم ست عشري يوازي أربعة أرقام ثنائية بالضبط، وبالتالي يمكن استخدام الثماني والست عشري كاختزال لعرض القيم الثنائية.

أنواع المعطيات

المعطيات العددية

تكون الكمبيوترات في قمة كفاءتها عند عملها في النظام الثنائي وحده لأن الأعداد الثنائية مناسبة للغاية للأجهزة الالكترونية، ويصمم الكمبيوتر النموذجي على أساس وحدة أساسية من المعطيات الثنائية تُسمى كلمة (تتألف من 8 أو 16 أو 32 خوينة عادة)، وتضبط الخوينة ذات المنزلة الكبرى الى جانبها لتحتجز اشارة (0 لـ+ و1 لـ−)، أما الخوينات المتبقية فهي خوينات معطيات. وليس هناك من مرفق لتوفير الفاصلة العشرية، فتخطيطها من مسؤولية المبرمج. على سبيل المثال، ان أكبر قيمة ثنائية يمكن خزنها في كمبيوتر تتألف كلمته من 32 خوينة هي:

01111111111111111111111111111111

وهي 2,147,483,647 في العشري، في حين أن حدود آلة ست عشرية الخوينات هي:

0111111111111111

وهي 32,767 في العشري.

وتناسب الأرقام الثنائية العديد من الاستخدامات، لكن هناك حاجة لأعداد كبيرة وأعداد صغيرة وأعداد كسرية في بعض الأحيان. في التمثيل العلمي تكتب الأعداد في شكل كسر عشري يتبعه أس للعدد 10، على سبيل المثال، ان سرعة الضوء هي 186,000 ميل في الثانية، فتكتب في شكل 0.186×10^6. ويستطيع العديد من الكمبيوترات خزن التقريبات الثنائية للأعداد العلمية التي تسمى بالأعداد الحقيقية أو الطليقة الفاصلة ومعالجتها.

وتتطلب بعض التطبيقات، وخصوصاً التطبيقات المتعلقة بالأعمال، أعداداً عشرية مدورة بدقة، وفي حين يناسب أي نوع من المعطيات الأعداد التامة أو الأرقام، فان الأعداد الطليقة الفاصلة والأعداد الثنائية تعطي تقريباً قريباً جداً للكسور العشرية في أفضل الأحوال، وبالتالي فان معظم الكمبيوترات يحتمل نوعاً من المعطيات العشرية. وعموماً تكون الكمبيوترات عند الحد الأدنى لكفاءتها عند معالجتها معطيات عشرية.

المعطيات النضيدية

لا تتوقف الكمبيوترات عند خزن ومعالجة الأعداد، غير ان الكثير من الاستخدامات يستدعي معطيات كالأسماء والعناوين ومواصفات المنتجات. وعادة تخزن هذه القيم النضيدية كمجموعات من السمات أو الرموز المنفردة، وتمثل كل سمة بكود معين ككودي الآسكي (ASCII) أو الأبسديك (EBCIDIC) اللذين ورد الحديث عنهما في الفصل الثاني (أنظر الشكل 2.6). وفي العديد من الكمبيوترات يشغل الرمز المكود الواحد خانة واحدة، وبالتالي سيخزن الاسم «يوحنا» في خمس خانات متعاقبة.

ومن المهم أن نلاحظ أن النضائد والأعداد أمران مختلفان، على سبيل المثال، اذا طبعت الرقم 1 متبوعاً بالرقم 2 فسيخزن كل رمز كنضيد من خانة واحدة في الذاكرة الرئيسية. وفي كمبيوتر يستخدم كود الآسكي سيظهر هذان الرمزان على الشكل التالي:

01010001 01010010

وليس هذا بالعدد 12، ففي كمبيوتر ست عشري الخوينات يخزن العدد الثنائي الصرف 12 على شكل:

0000000000001100

(جرب استعمال قاعدة الرقم ضرب قيمة المنزلة). إن الأعداد والنضائد شيئان مختلفان، وهذا هو السبب الذي يحملك على تمييز النضائد عن الأعداد في معظم لغات البرمجة. ان القيمة الموضعية لكل رقم في عدد ما هامة، فلا معنى للقيمة الموضعية للخوينات المنفردة في نضيد ونحن ننتقل من خانة الى خانة.

تدخل المعطيات الكمبيوتر عادة عبر جهاز دخل في شكل سلسلة، ولمعظم الكمبيوترات تعليمات خاصة لتحويل السلسلات الى أعداد، فالعمليات الحسابية تؤدى على الأعداد، وتحول الأعداد مجدداً الى شكل سلسلة قبل أن ترسل الى جهاز خرج. ويؤدي لك معظم لغات البرمجة تحويلات أنواع المعطيات هذه، أما لغات المترجم الجامع فهي استثناء لهذه القاعدة.

مسرد

يحتوي هذا المسرد على تعريفات قصيرة قصد بها أن تنقل الى القارئ شرحاً موجزاً لمعاني المصطلحات الأساسية المختارة. وللحصول على تعريفات أكثر دقة أنظر:

المعجم الأميركي الوطني لمعالجة المعلومات الذي نشره المعهد الأميركي الوطني للمعايير، 1430 شارع برودواي، نيويورك ولاية نيويورك 10018. The American National Dictionary for Information Processing, American National Standards Institute, 1430 Broadway, New York, New York 10018.

معجم مصطلحات معالجة المعطيات الخاص بمنظمة المعايير العالمية The ISO Vocabulary of Data Processing, by The International Standards Organization الذي نشرته المنظمة نفسها.

آلية النيل **Access mechanism**
الجزء الذي يحمل رأس القراءة/الكتابة على القرص. تتحرك آلية النيل لتضع رأس القراءة/الكتابة في موضعه على السكة التي تحتوي على المعطيات المطلوبة، تماماً كما يفعل ذراع الفونغراف على القرص الدوار الذي يحمل أسطوانة الفونغراف.

أسلوب النيل **Access method**
نهج كيان منطقي يترجم طلب المبرمج لدخل أو خرج الى الأوامر المادية التي يتطلبها الجهاز الخارجي.

عنوان **Address**
موقع في الذاكرة. وغالباً ما ترقم الخانات أو الكلمات التي تشكل الذاكرة بشكل تسلسلي، ويكون رقم الخانة (أو الكلمة) هو عنوانها.

الخوارزمية **Algorithm**
قانون أو مجموعة قوانين للوصول الى إجابة بعدد محدود من الخطوات.

إشارة قياسية، نظيري **Analog**
معطيات تمثل بشكل مادي مستمر. إن ارتفاع عمود الزئبق هو تمثيل نظيري لدرجة حرارة

ما. تنقل معطيات الكمبيوتر على خطوط التلفونات المحلية وذلك بتحويلها الى شكل موجي مستمر. قارن بـDigital.

التحليل Analysis
تلك الخطوة في تحليل الأنظمة وعملية التصميم التي يحدد محلل الأنظمة في أثنائها ما يجب أن يجرى لحل مشكلة ويضع خطة نظام منطقي.

برنامج تطبيقي Application program
برنامج يكتب لأداء مهمة مستخدم طرفي. ان برامج جداول الرواتب أو العاب الكمبيوتر هي برامج تطبيقية، أما نظام التشغيل فليس برنامجاً تطبيقياً.

Architecture
أنظر «Computer architecture».

وحدة الحساب والمنطق Arithmetic and logic unit
ذلك الجزء من معالج الكمبيوتر الذي ينفذ التعليمات.

صفيف Array
بنية معطيات توزع فيها الذاكرة على شكل سلسلة من الخلايا المرقمة. يمكن وضع عناصر معطيات فردية أو استخراجها من واحدة من هذه الخلايا بالرجوع الى رقم أو أرقام الخلية.

المترجم الجامع Assembler
لغة برمجة تكود فيها عبارة مصدرية مختصرة واحدة لكل تعليمة على مستوى الآلة.

احتياطي Backup
كيان مادي أو كيان منطقي اضافي أو معطيات اضافية يقصد بها إبقاء نظام الكمبيوتر عاملاً في حال تعطل مكون واحد أو أكثر.

أساس، قاعدة Base
في نظام عددي، العدد الذي يستخدم لتحديد قيم موقعية. على سبيل المثال، يستعمل النظام العشري الأساس 10 في حين يستعمل الثنائي العدد 2 كأساس له.

معالجة دفعية Batch processing
نوع من تطبيقات الكمبيوتر تجمع فيه المعطيات مع الوقت ثم تعالج بعضها مع بعض. على سبيل المثال، يمكن أن تجمع معطيات جدول الرواتب طوال الأسبوع وتعالج يوم الجمعة. قارن بـTransaction processing.

بود Baud
الوحدة الأساسية لسرعة الاتصالات المعطياتية وتقاس بعدد مرات حدوث اشارة متميزة في الثانية. وبما أن الوحدة الأساسية للمعطيات المرسلة هي الخوينة، فان سرعة بود تقيس عدد الخوينات في الثانية.

ثنائي Binary
نظام عددي أساسه 2 يستعمل القيمتين 0 و 1.

خوينة Bit
رقم ثنائي.

Board
أنظر «Circuit board»

تحميل Boot
نهج صغير يقرأ الى الذاكرة الرئيسية عندما يشغل الكمبيوتر، وهو يقرأ بقية نظام التشغيل الى الذاكرة. أنظر أيضاً Initial program load.

وسيط Buffer
خزن موقت يستعمل ليعوض عن السرعات المختلفة للأجهزة المتجاورة.

شائبة، خطأ Bug
خطأ في برنامج.

ناقل Bus
مجموعة أسلاك متوازية تستخدم لارسال المعطيات أو الأوامر أو الطاقة.

خانة Byte
ثمان خوينات. هي في كثير من الأنظمة الكمبيوترية، أصغر وحدة يمكن مخاطبتها في الذاكرة الرئيسية.

كبل Cable
موصل كهربائي، وهو غالباً سلك تسلسلي مغلف.

إشارة توصيل Carrier signal
اشارة معيارية يمكن التنبؤ بها، مثل موجة سينية تستعمل لارسال المعطيات. وتشير الاشارة غير المعدلة الى انعدام المعطيات، وتؤدي التعديلات في النمط المعياري المعنى.

كاسيت أو كاسيت مغنطيسي Cassette or magnetic cassette
وسط خزن مساعد رخيص الثمن. شريط الكاسيت العادي.

Central processing unit
أنظر «Processor».

قناة Channel
جهاز يستعمل لالحاق أجهزة دخل وخرج وخزن مساعد بنظام كمبيوتر كبير. تحتوي

القناة على معالجها الخاص، وبالتالي فهي تستطيع أن تحرر المعالج الرئيسي من مسؤولية التحكم في عمليات الدخل/الخرج.

سمة، رمز **Character**
حرف أو رقم واحد، أو رمز آخر. وفي كثير من الكمبيوترات، تستطيع كل خانة أن تحتجز رمزاً في شكل مكود.

رقيقة **Chip**
مربع متناهي الصغر من السيليكون يحمل الآلاف من الدارات الالكترونية المتكاملة.

لوحة دارات **Circuit board**
سطح مستو توصل عليه الرقيقات بمسارات الكترونية مدمجة في السطح. تشمل الأمثلة عليها لوحات المعالج ولوحات الذاكرة واللوحات البينية.

موقت **Clock**
جهاز يولد نبضات الكترونية منتظمة توجّه الكمبيوتر.

كود **Code**
(1) مجموعة من القوانين لتمثيل السمات على شكل أنماط خوينية. (2) كتابة برنامج.

أمر **Command**
(1) إشارة تحكم تأمر مكون كيان مادي أن يؤدي وظيفة محددة. على سبيل المثال، يأمر أمر استحضار الذاكرة بنقل محتويات موقع ذاكرة واحد الى ناقل، في حين يأمر أمر نشد بينية قرص بأن تحدد موقع آلية النيل. (2) طلب من مبرمج أو مشغل أو مستخدم الى نظام تشغيل لأداء وظيفة محددة كطلب تحميل برنامج.

لغة أمر **Command language**
لغة للاتصال مع نظام تشغيل.

معالج اوامر **Command processor**
وحدة نظام التشغيل التي تقرأ الأوامر وتفسرها وتنفذها.

بينية اتصالات **Communication interface**
جهاز يوصل خط اتصالات مع نظام كمبيوتر.

مصرف، برنامج مترجم **Compiler**
برنامج مساند يقرأ برنامجاً مصدرياً ويترجم العبارات المصدرية الى لغة الآلة ويخرج برنامجاً تجميعياً ثنائياً كاملاً.

الكمبيوتر **Computer**
آلة تعالج المعطيات وتحولها الى معلومات تحت تحكم برنامج مخزون.

تصميم بنيوي للكمبيوتر **Computer architecture**
التركيب المادي للكمبيوتر، وبشكل خاص الطريقة التي توصل بها مكونات الكمبيوتر بعضها مع بعض.

برنامج الكمبيوتر **Computer program**
سلسلة من التعليمات التي توجه الكمبيوتر خلال معالجة ما.

متزامن **Concurrent**
في الفترة الزمنية نفسها.

كونسول **Console**
جهاز ما، وهو غالباً وحدة عرض ولوحة مفاتيح، يتصل العامل من خلالهما مع الكمبيوتر.

مجاور **Contiguous**
ملاصق، محاذ.

مستمر **Continuous**
غير منقطع؛ متصل. قارن بـDiscrete.

Control unit, instruction
انظر «Instruction control unit».

وحدة تحكم، دخل/خرج **Control unit, I/O**
جهاز الكتروني يصل جهاز دخل/خرج بقناة.

CPU
لفظة أوائلية لـ«Central processing unit».

الزليقة **Cursor**
السطر الوامض أو الاطار الذي يشير الى المكان الذي سيظهر فيه الرمز التالي الذي سيطبع أو يخرج على شاشة عرض.

دورة **Cycle**
أية مجموعة عمليات تكرّر بانتظام وبالترتيب نفسه. وفي الاتصالات المعطياتية، تمثل موجة سينية واحدة كاملة (S على طرفها) دورة واحدة.

أسطوانة **Cylinder**
على رزمة أقراص متعددة الأسطح، موقع واحد من مواقع آلية النيل يحدد مجموعة من عدة سكك.

المعطيات **Data**
حقائق خام غير مركبة وغير معالجة.

قاعدة المعطيات **Database**
مجموعة من المعطيات ذات الصلة. وهي عموماً مجموعة مركزية متكاملة من معطيات إحدى المنظمات.

نظام ادارة قاعدة المعطيات Database management system
كيان منطقي و/أو كيان مادي يتحكم في نيل قاعدة معطيات.

اتصالات معطياتية Data communication
ارسال المعطيات عبر خط اتصالات.

برنامج مراقبة الاتصالات المعطياتية Data communication monitor
نهج برامجي أو برنامج يتحكم في عملية الاتصالات المعطياتية أو يراقبها.

تبعية للمعطيات Data dependency
حالة تحدث عندما يعتمد منطق برنامج ما الى حد بعيد على البنية المادية لمعطياته. من الصعب صيانة البرامج التي تعتمد على المعطيات.

معجم معطيات Data dictionary
مجموعة معطيات حول المعطيات التي يعالجها نظام ما.

عنصر معطيات Data element
وحدة معطيات واحدة ذات معنى.

مخطط سير المعطيات Data flow diagram
تمثيل بياني لنظام منطقي يظهر كيفية تدفق المعطيات بين المصادر، والمعالجات ووحدات الخزن وأهدافها.

ادارة المعطيات Data management
خزن المعطيات بطريقة يمكن معها استرجاعها عند الحاجة.

معالجة المعطيات Data processing
تحويل المعطيات الى معلومات.

Data set
أنظر «Modem».

بنية معطيات Data structure
مجموعة منظمة من المعطيات، ومن الأمثلة على ذلك لائحة وصفيف وملف.

كشف الخطأ وتصحيحه Debug
إزالة الأخطاء (الشوائب) من البرنامج.

لغة اعلانية Declarative language
أنظر Nonprocedural language.

إزالة التضمين Demodulation
التضمين المعكوس. تحويل المعطيات من شكل نظيري الى شكل رقمي.

تصميم Design
خطوة في تحليل الأنظمة وعملية التصميم يطور المحلل خلالها تصميماً مادياً للنظام.

رقمي Digital
معطيات تمثل على شكل أرقام مستقلة متميزة. قارن بـAnalog.

نيل مباشر Direct access
نيل المعطيات دون إعارة الانتباه لموضعها المادي في ملف أو قرص. قارن بـSequential access.

الدليل Directory
أنظر Index. عموماً يستعمل المصطلح «دليل» ليعني لائحة الملفات المخزونة في قرص أو وسيط ثانوي آخر. أمّا المصطلح Index فهو أكثر شمولية.

متقطع، متميز Discrete
عكس مستمر. منفصل، مستقل.

قرص مغنطيسي Disk, magnetic
سطح مستو يشبه القرص يمكن خزن المعطيات عليه مغنطيسياً.

مجموعة اقراص Disk pack
مجموعة من قرصين أو اكثر مكدسة على عمود مدوار مشترك، ويتم نيلها بواسطة رؤوس قراءة/كتابة.

قريص مرن Diskette
قرص مغنطيسي مرن رقيق يُستعمل غالباً في أنظمة الكمبيوتر الصغيرة.

عرض أو شاشة عرض Display or display screen
شاشة تشبه شاشة التلفزيون تعرض المعطيات.

معالجة موزعة للمعطيات Distributed data processing
أسلوب لاستعمال الكمبيوترات التي تستخدم فيها آلات عديدة لنشاطات المعالجة المحلية للمعطيات وهي موصولة بكمبيوتر مركزي يحتوي غالباً قاعدة المعطيات الخاصة بالمؤسسة. ويقدم مثل هذا التشكيل كثيراً من فوائد اللامركزية دون التضحية بفوائد المركزية.

التوثيق Documentation
رسوم بيانية وتعليقات وغيرها من المواد التي تشرح برنامجاً أو توضحه.

ادارة الذاكرة الدينامية Dynamic memory management
تخصيص فسحة الذاكرة للبرامج التطبيقية كلما نشأت الحاجة لتلك الفسحة.

تنفيذ Execution
عملية تنفيذ تعليمة أو تأدية نهج.

زمن التنفيذ **Execution time or E-time**
الوقت الذي تنفذ خلاله تعليمة بواسطة وحدة الحساب والمنطق.

دراسة جدوى **Feasibility study**
دراسة تنجز في مرحلة مبكرة من عملية تحليل النظام وتصميمه وتهدف الى تحديد ما إذا كان من الممكن حل المسألة أم لا.

استحضار **Fetch**
تحديد موقع وجدة معطيات أو تعليمة في الذاكرة الرئيسية وإرسالها على ناقل الى المعالج.

بصريات الألياف **Fiber optics**
وسيط للاتصالات المعطياتية تنقل بواسطته نبضات ضوء ليزري عبر كبل من الألياف الزجاجية.

حقل **Field**
عنصر معطيات واحد ذو معنى في ملف.

ملف **File**
مجموعة من السجلات ذات العلاقة.

ادارة الذاكرة بالتقسيم الثابت **Fixed partition memory management**
أسلوب لادارة الذاكرة تقسم فيه فسحة الذاكرة المتوافرة الى أقسام عديدة محددة الطول ويحمل برنامج واحد في كل قسم.

Floppy disk
أنظر «Diskette».

مخطط سير العمليات **Flowchart**
تمثيل بياني لبرنامج تمثل الرموز فيه الخطوات المنطقية وتحدد خطوط التدفق تسلسل تلك الخطوات.

Fourth-generation language
أنظر «Nonprocedural language».

جهاز أمامي **Front-end device**
وحدة تحكم في الارسال تصل عدداً من خطوط الاتصال بنظام كمبيوتر. وغالباً ما يحتوي الجهاز الأمامي على قدر كاف من الذكاء يمكنه من تولي مسؤولية استطلاع الطرفيات.

تخطيطيات **Graphics**
خرج كمبيوتر على شكل نقط وخطوط وأشكال.

قرص صلب **Hard disk**
قرص صلد. قارن بـFloppy disk أو Diskette. وعموماً يدور القرص الصلب دائماً، ونتيجة لذلك فإن نيل المعطيات من القرص الصلب أسرع بكثير منه في القريص. وللقرص الصلب أيضاً سعة خزن معطيات أكبر.

كيان مادي **Hardware**
معدات مادية. قارن بـSoftware.

مخطط تسلسل هرمي **Hierarchy chart**
أداة لوضع نموذج لبرنامج كمبيوتر على شكل تسلسل هرمي من الوحدات الأحادية الوظيفة.

التنفيذ **Implementation**
تلك الخطوة في عملية تحليل وتصميم الأنظمة التي تكتب في أثنائها البرامج ويختار الكيان المادي ويركب وتكتب اجراءات التشغيل الخ. وينتهي التنفيذ باتاحة البرنامج أو النظام لمستخدم ما.

الفهرس **Index**
لائحة بمحتويات مجموعة أقراص أو أي وسيط خزن آخر تظهر موقع كل ملف أو برنامج. وبصورة أعم، تستعمل قيمة أو مجموعة قيم مثل مرصف فهرسي أو رمز دليلي لتحديد موقع عناصر معطيات معينة.

معلومات **Information**
المعنى الذي يخصّصه الانسان للمعطيات. معطيات تمّت معالجتها.

Information processing
أنظر «Data processing».

تحميل بدئي **Initial program load or IPL**
عملية تحميل نظام التشغيل عندما يبدأ تشغيل الكمبيوتر لأول مرة. تحمل أنظمة الكمبيوتر الكبيرة والصغيرة تحميلاً بدئياً.

دخل **Input**
تحويل المعطيات من جهاز خارجي الى ذاكرة كمبيوتر رئيسية.

نظام التحكم بالدخل/الخرج **Input/output control system or IOCS**
وحدة نظام التشغيل التي تتولى مسؤولية الاتصال مباشرة بأجهزة الدخل والخرج والخزن المساعد.

تعليمة **Instruction**
خطوة واحدة في برنامج ما. وتأمر كل تعليمة الكمبيوتر بتأدية واحدة من وظائفه الأساسية.

وحدة التحكم بالتعليمات **Instruction control unit**
ذلك الجزء من معالج الكمبيوتر الذي يقرّر أية تعليمة ستنفذ لاحقاً.

عداد التعليمات **Instruction counter**
مرصف خاص يحتجز عنوان التعليمة التالية التي سيتم تنفيذها.

مرصف تعليمة **Instruction register**
مرصف خاص يحتجز التعليمة التي يجري تنفيذها بواسطة المعالج.

مجموعة تعليمات **Instruction set**
الدارات الالكترونية التي تجمع وتطرح وتضرب وتقسم وتنسخ وتقارن وتطلب الدخل والخرج. وهذه الدارات هي التي تشكل وحدة الحساب والمنطق في معظم الكمبيوترات.

زمن التعليمة **Instruction time or I-time**
الزمن الذي تستحضر في أثنائه التعليمة التالية من الذاكرة الرئيسية وتفسر بواسطة وحدة التحكم بالتعليمات الخاصة بالمعالج.

ذاكرة دارات متكاملة **Integrated circuit memory**
ذاكرة مكونة من رقيقات دارات متكاملة.

بينية **Interface**
في كمبيوتر صغير، مكون الكتروني، غالباً ما يكون لوحة، يصل جهازاً خارجياً بالكمبيوتر. وبصورة أعم، هو مكون الكتروني يصل بين جهازين مختلفين.

مفسر **Interpreter**
برنامج مساند يقرأ عبارة مصدرية واحدة ويترجم تلك العبارة الى لغة الآلة وينفذ تلك التعليمات على مستوى الآلة ومن ثم ينتقل الى العبارة المصدرية التالية. قارن بـCompiler.

انقطاع **Interrupt**
إشارة الكترونية تتسبّب في توقف الكمبيوتر عن العمل الذي يقوم به وتنقل التحكم الى نظام التشغيل بطريقة تجعل من الممكن لاحقاً استئناف تنفيذ المهمة التي توقف الكمبيوتر عن ادائها لحظة الانقطاع.

I/O
اختصار لـ«Input/output».

I/O control unit
انظر «Control unit, I/O».

تخصيص جهاز دخل/خرج **I/O device allocation**
مهمة (يؤديها نظام إلتشغيل عادة) تخصيص اجهزة الدخل/الخرج والخزن المساعد للبرامج التطبيقية.

IPL
انظر «Initial program load».

كيلوبايت **K**
عندما نتحدث عن سعة الذاكرة أو الخزن المساعد فإن كيلوبايت تعني 1024 خانة أو كلمة.

لوحة مفاتيح **Keyboard**
جهاز دخل تمثل السمات عليه كمفاتيح مستقلة. وعندما يضغط على مفتاح، تدخل السمة المرتبطة به الى نظام الكمبيوتر.

مكتبة **Library**
مجموعة من الملفات المتعلق بعضها ببعض.

خط اتصال **Line**
وسيط اتصال يصل بين نقطتين أو أكثر.

محرر الوصل **Linkage editor**
برنامج نظام يجمع الوحدات التجميعية لتشكل وحدة تحميل، ويخرج وحدة التحميل الى مكتبة، ومن ثم يحمل البرنامج في الذاكرة الرئيسية.

بنية اللائحة **List structure**
لائحة من عناصر تفصل بينها فواصل وعلامات تنقيط أو أي رمز فصل آخر.

وحدة تحميل **Load module**
برنامج كامل على مستوى الآلة في شكل جاهز للتحميل في الذاكرة الرئيسية والتنفيذ.

محمل **Loader**
برنامج نظام يجمع الوحدات التجميعية لتشكل وحدة تحميل ثم يحمل البرنامج في الذاكرة الرئيسية. وهو شبيه بمحرر الوصل، غير أن المحمل لا يخرج وحدة التحميل إلى المكتبة.

موضعي **Local**
موصول إلى الكمبيوتر بأسلاك كهربائية عادية. قريب جداً من الكمبيوتر. قارن بـRemote.

دخل/خرج منطقي **Logical I/O**
عمليات دخل أو خرج تؤدى دون اعتبار للبنية المادية للمعطيات. طلب لسجل منطقي في ظل الادارة التقليدية للمعطيات.

نظام منطقي **Logical system**
تصميم نظام يركز على ما يجب عمله ولكنه لا يركز على كيفية عمله.

دورة الآلة **Machine cycle**
دورة تشغيل المعالج الأساسية التي يتم في اثنائها استحضار تعليمة واحدة وتفسيرها وتنفيذها.

لغة الآلة **Machine language**
تعليمات ثنائية يمكن للكمبيوتر ان يخزّنها في الذاكرة الرئيسية، وان يستحضرها وينفذها.

Magnetic disk
أنظر «Disk, magnetic».

اسطوانة مغنطيسية **Magnetic drum**
اسطوانة مطلية بمادة مغنطيسية على سطحها

الخارجي. تخزن المعطيات حول السطح الخارجي ويتم نيلها/بسلسلة من رؤوس القراءة/الكتابة، كل رأس منها مخصص لسكة. وسيط سريع جداً له سعة خزن محدودة. أول وسيط خزن مساعد ذو أهمية.

وسط مغنطيسي **Magnetic media**
وسط خزن مساعد أو دخل أو خرج يمثل المعطيات كأنماط مغنطيسية.

شريط مغنطيسي **Magnetic tape**
وسيط مساند واسع الانتشار. شريط من الميلار مطلي بمادة مغنطيسية، وتسجل المعطيات على طول سطح الشريط.

كمبيوتر رئيسي **Mainframe**
وحدة المعالجة في نظام كمبيوتر كبير، أو وحدة المعالجة بالاضافة الى مكونات أخرى موجودة في الخزانة المادية نفسها كوحدة معالجة نظام كمبيوتر كبير.

ذاكرة رئيسية أو خزن رئيسي **Main memory or main storage**
الذاكرة التي يمكن أن ينالها المعالج مباشرة.

معالج رئيسي **Main processor**
المعالج الرئيسي في كمبيوتر معالجة متعددة. وفي بعض الأحيان يستعمل المصطلح كرديف لـ Processor.

صيانة **Maintenance**
مساندة مستمرة لبرنامج أو نظام بعد بيعه.

ذاكرة **Memory**
مكون الكمبيوتر الذي تخزن فيه التعليمات والمعطيات.

ادارة الذاكرة **Memory management**
تخصيص فسحة الذاكرة الرئيسية للبرامج التطبيقية، المهمة التي يؤديها نظام التشغيل عادة.

تحويل التبليغة **Message-switching**
تسيير التبليغات وذلك بتسلمها وخزنها وارسالها.

ميكروكمبيوتر **Microcomputer**
نظام كمبيوتر صغير. يتكون الميكروكمبيوتر النموذجي من معالج ميكروي، وذاكرة رئيسية وجهاز دخل/خرج واحد أو اكثر.

معالج ميكروي **Microprocessor**
معالج على رقيقة دارة متكاملة واحدة. المعالج في نظام ميكروكمبيوتر.

ميكروثانية **Microsecond**
واحد على مليون من الثانية.

موجة ميكروية **Microwave**
موجة الكترومغنطيسية تستعمل لارسال المعطيات.

ملي ثانية **Millisecond**
واحد على ألف من الثانية.

مينيكمبيوتر **Minicomputer**
كمبيوتر رقمي صغير أصغر من الكمبيوتر الرئيسي، ولكنه أكبر من الميكروكمبيوتر.

مودم **Modem**
لفظة أوائلية لـMOdulator DEModulator. جهاز يحول المعطيات من شكلها الرقمي في داخل الكمبيوتر الى الشكل الموجي النظيري ثم يعيدها سيرتها الأولى مرة أخرى، ويستعمل لوصل معدات الكمبيوتر بخط تلفون.

تضمين **Modulation**
تحويل المعطيات من الشكل الرقمي الى الشكل الموجي النظيري.

Monitor
أنظر «Display».

اللوحة الأم **Motherboard**
هيكل معدني أو بلاستيكي يحمل لوحات دارات الكمبيوتر. وغالباً تنزلق لوحات الدارات في شقوق على الهيكل، وتوصل الكترونياً بواسطة ناقل.

أم اس - دوس **MS-DOS**
نظام تشغيل ميكروكمبيوتر واسع الانتشار.

تصميم بنيوي متعدد النواقل **Multiple-bus architecture**
نظام بنيوي للكمبيوتر يستخدم فيه أكثر من خط ناقل واحد لوصل المكونات. وغالباً ما توفر خطوط نواقل منفصلة للأوامر والعناوين والمعطيات، وفي أغلب الأحيان تصل خطوط نواقل مستقلة المعالج الرئيسي بالذاكرة الرئيسية، ومعالجات القنوات بالمعالج الرئيسي، وذاكرات القنوات بالذاكرة الرئيسية.

معالجة متعددة **Multiprocessing**
معالجان مستقلان، أو اكثر، يتشاركان ذاكرة واحدة.

البرمجة المتعددة **Multiprogramming**
كمبيوتر واحد ينفذ عدة برامج بصورة متزامنة.

نانوثانية **Nanosecond**
واحد على بليون من الثانية.

شبكة **Network**
كمبيوتران، أو أكثر، موصولان بخطوط اتصالات.

تشويش **Noise**
في الاتصالات المعطياتية، تداخل الكتروني.

لغة غير اجرائية **Nonprocedural language**
لغة برمجة يصف فيها المبرمج بكل بساطة البنية المنطقية لمشكلة ما بدلاً من كتابة إجراء لحلها. وهي تسمى ايضاً بلغة الجيل الرابع أو اللغة الاعلانية.

ذاكرة مستقرة **Nonvolatile memory**
الذاكرة التي تحافظ على محتوياتها حتى بعد فقدان الطاقة. ان الخزن الثانوي هو ذاكرة مستقرة أما ذاكرة النيل العشوائي (RAM) فهي ذاكرة غير مستقرة عادة.

وحدة تجميعية **Object module**
ترجمة على مستوى الآلة للكود المصدري الخاص بالمبرمج.

مفتوح **Open**
تحضير ملف للمعالجة. على سبيل المثال، يشتمل فتح ملف في قرص على فحص الفهرس لايجاد السكة والقطاعات حيث تكون معطيات الملف مخزونة.

معامل **Operand**
ذلك الجزء من التعليمة الذي يعين المراصف و/أو مواقع الذاكرة التي ستشترك في العملية.

نظام التشغيل **Operating system**
مجموعة من وحدات البرنامج التي تتحكم في عمل الكمبيوتر. ويحدد نظام التشغيل النموذجي الموارد ويجدول البرامج ويتحكم في نيل أجهزة الدخل والخرج ويدير المعطيات.

كود عملية **Operation code**
ذلك الجزء من التعليمة الذي يعين العملية التي ستنجز، على سبيل المثال، إجمع او اطرح الخ...

وسط بصري **Optical media**
عموماً، وسط دخل تمثل فيه المعطيات كأنماط ساطعة وقاتمة يمكن أن يفسرها ماسح بصري.

خرج **Output**
فعل نقل المعطيات أو المعلومات من ذاكرة الكمبيوتر الرئيسية الى جهاز خارجي.

كيان منطقي مجمع **Packaged software**
برامج مشتراة. برامج تمّ الحصول عليها في شكل نهائي جاهز للاستعمال. وهي على عكس الكيان المنطقي المعدل وفقاً لطلب العميل والكيان المنطقي الأصلي.

تحويل الربطة **Packet switching**
أسلوب لارسال المعطيات تقسم فيه التبليغة الى مجموعات رقمية منفصلة، ثم ترسل المجموعات مستقلاً بعضها عن بعض على خط فائق السرعة ويُعاد تجميع التبليغة في الطرف الآخر.

متواز **Parallel**
جنباً الى جنب. وتشتمل المعالجة المتوازية على اداء مهمتين أو اكثر في الوقت نفسه بينما يشتمل ارسال المعطيات المتوازي على ارسال الخوينات جنباً الى جنب على أسلاك متوازية. قارن بـSerial.

خوينة التكافؤ **Parity bit**
خوينة اضافية في الذاكرة تلحق بخوينات المعطيات وتسمح للكمبيوتر بفحص النمط الخويني توخياً للدقة.

كيان مادي محيطي **Peripheral hardware**
اجهزة دخل وخرج وخزن مساعد متصلة بنظام كمبيوتر.

كمبيوتر شخصي **Personal computer**
نظام كمبيوتر صغير غير مكلف يسوق لاستعمال الأفراد.

دخل/خرج مادي **Physical I/O**
فعل تحويل مجموعة من المعطيات المادية من جهاز محيطي أو اليه. على سبيل المثال قد تحول كل عملية دخل/خرج مادي على قريص قطاعاً واحداً، وفي الطابعة، قد تنقل كل عملية دخل/خرج مادي سطراً واحداً.

نظام مادي **Physical system**
تصميم نظام يحدد مكونات مادية معينة مثل برامج الكمبيوتر وأجهزة الدخل والخرج وأجهزة الخزن المساعد وغيرها من الكيانات المادية وملفات المعطيات وتدفقات المعطيات والوسائط والاجراءات.

بيكو ثانية **Picosecond**
واحد على مليون من واحد على مليون من الثانية.

بيكسل **Pixel**
عنصر صورة. نقطة على شاشة عرض يمكن أن تُضاء أو تُطفأ اختيارياً. الوحدة الأساسية في عرض تخطيطي.

استطلاع **Polling**
سؤال سلسلة من الطرفيات أو فحص سلسلة

من الوسائط واحداً بعد الآخر لمعرفة إذا ما كان لديها معطيات للارسال. أسلوب لتحديد (الجهاز) الذي عليه أن يرسل المعطيات في المرة التالية.

Port بوابة/مدخل
النقطة التي تدخل منها خطوط الاتصالات نظام كمبيوتر.

Primitive command أمر أولي/بدائي
أمر على مستوى الآلة لكيان مادي، على سبيل المثال، أمر استحضار يطلب المعالج بموجبه المعطيات من الذاكرة الرئيسية أو نشد تؤمر بينية قرص بموجبه بوضع آلية النيل في موضع معين.

Printer طابعة
جهاز يخرج سمات أو رموزاً مطبوعة.

Problem definition تحديد المشكلة
الخطوة الأولى في عملية البرمجة و/أو تحليل الأنظمة وتصميمها التي يحدد المبرمج أو المحلل في أثنائها المشكلة.

Processing معالجة
(1) تنفيذ تعليمات. (2) تحويل معطيات الى معلومات.

Processor معالج
ذلك المكون في الكمبيوتر الذي يختار التعليمات وينفذها. يحتوي المعالج على موقت ووحدة تحكم في التعليمات ووحدة حساب ومنطق ومراصف.

Processor management ادارة المعالج
مهمة توزيع وقت المعالج على البرامج التطبيقية، ويؤدي هذه المهمة عادة نظام التشغيل.

Program
أنظر «Computer program».

Programmer مبرمج
شخص يكتب برامج الكمبيوتر.

PROM
ذاكرة قراءة فقط قابلة للبرمجة.

Prompt طلب إدخال
تبليغة قصيرة يطبعها أو يعرضها البرنامج أو نظام التشغيل تطلب من المستخدم أن يقدم دخلاً.

Protocol بروتوكول
مجموعة قوانين لاقامة اتصال بين جهازين.

Punched card بطاقة مثقبة
وسط دخل يمثل المعطيات على شكل انماط من ثقوب مخرمة على بطاقة.

Queueing صف
وضع البرامج التطبيقية على خط (أو صف) انتظار تمهيداً لتحميلها في الذاكرة الرئيسية.

RAM (random access memory) ذاكرة نيل عشوائي
ذاكرة يمكن للمبرمج عنونتها وقراءتها وكتابتها مباشرة. ان ذاكرة الكمبيوتر الرئيسية هي عموماً ذاكرة نيل عشوائي. قارن بـROM.

Random access
أنظر «Direct access».

Read/write head راس قراءة/كتابة
المكون الذي يحول المعطيات من سطح القرص أو الشريط المغنطيسي أو اليه.

Record سجل
مجموعة من الحقول ذات العلاقة. على سبيل المثال، كل الحقول المتعلقة براتب الموظف ستكون سجل جدول راتب ذلك الموظف، وكل الحقول المتعلقة بالأداء الأكاديمي لطالب ما ستكون سجل التاريخ الأكاديمي لذلك الطالب.

Redundant data معطيات متكررة/مطنبة
معطيات تتكرر في مكانين أو اكثر.

Register مرصف، مركم
خزن موقت يستعمل لاحتجاز المعطيات، أو التعليمات أو معلومات التحكم، في المعالج. وعادة تخزن في المراصف التعليمة الحالية والمعطيات التي تجري معالجتها بتلك التعليمة ومعلومات التحكم الأساسية.

Relative address عنوان نسبي
عنوان منسوب لنقطة مرجعية، على سبيل المثال، الخانة العاشرة من بداية برنامج ما، أو السجل الثالث في ملف ما.

Relative record number رقم السجل النسبي
موقع سجل بالنسبة لبداية ملف ما، على سبيل المثال، السجل الرابع في الملف. وإذا ما أعطينا السكة الفعلية والقطاع الخاصين بالسجل الأول في ملف ما، فمن الممكن احتساب عنوان أي سجل آخر اذا ما أعطينا رقم سجله النسبي.

Remote بعيد
ناء. موصول بكمبيوتر بواسطة خطوط اتصالات. قارن بـLocal.

Resolution وضوح
مقياس لدقة الصورة التخطيطية أو وضوحها، وغالباً دالة لعدد البيكسلات (عناصر الصورة) على شاشة ما.

ادارة الموارد **Resource management**
مسؤولية من مسؤوليات نظام التشغيل في كثير من أنظمة الكمبيوتر الكبيرة المتعددة البرمجة، وغالباً ما تشمل ادارة موارد معينة مثل وقت المعالج وفراغ الذاكرة الرئيسية ونيل أجهزة الخزن المساعد.

نقل الى الذاكرة/ نقل من الذاكرة **Roll-in/roll-out**
أسلوب لادارة الذاكرة يستخدم في أنظمة المشاركة الزمنية التي تنقل فيها البرامج الى خزن مساعد بين العمليات ثم تنقل مرة أخرى الى الذاكرة الرئيسية عندما تصل عملية ما.

ذاكرة قراءة فقط **ROM (read-only memory)**
نوع من الذاكرة لا يستطيع المبرمج أن يعدل فيه.

تاخير دوراني **Rotational delay**
الوقت المطلوب لدوران القطاع المطلوب على قرص أو أسطوانة الى راس القراءة/الكتابة بعد أن يكون الرأس قد وضع فوق السكة المطلوبة.

نهج، روتين **Routine**
وحدة برنامج.

جدولة **Scheduling**
تحديد أي من البرامج سيحمل في الذاكرة الرئيسية عندما يصبح الفراغ متيسراً، المهمة التي يؤديها نظام التشغيل عادة.

خزن مساعد **Secondary storage**
ذاكرة مستقرة، كالقرص أو الشريط المغنطيسي، تستعمل للخزن الطويل الأمد لتعليمات البرامج والمعطيات. وعادة تخزن التعليمات والمعطيات التي تجري معالجتها حالياً بواسطة الكمبيوتر في الذاكرة الرئيسية،، وتخزن كل المعطيات والتعليمات الأخرى في الخزن المساعد.

قطاع **Sector**
عنصر محدد الطول من القرص أو أي خزن مغنطيسي آخر يحتجز سجلاً مادياً واحداً. ويبلغ طول القطاع العادي في أنظمة الميكروكمبيوتر 512 سمة أو رمزاً.

حماية **Security**
عناصر كيان منطقي وكيان مادي وعناصر إجرائية يقصد منها وقاية أو حماية معدات الكمبيوتر ومعطياته أو برامجه.

زمن النشد **Seek time**
الوقت المطلوب لوضع آلية نيل قرص فوق سكة معينة.

نيل تسلسلي **Sequential access**
نيل السجلات بترتيب ثابت، هو عموماً الترتيب الذي سجلت به مادياً.

تسلسلي **Serial**
الواحد تلو الآخر. قارن بـParallel. ويشمل ارسال المعطيات التسلسلية ارسال دفق من الخوينات الواحدة تلو الأخرى على السلك نفسه.

تصميم بنيوي احادي الناقل **Single-bus architecture**
تصميم بنيوي للكمبيوتر تكون فيه كل المكونات الداخلية موصولة بخط ناقل واحد.

شق **Slot**
واحدة من عدة فتحات في اللوحة الأم يمكن ادخال قابس لوحة دارات فيها.

كيان منطقي **Software**
برامج. قارن بـHardware.

كود مصدري **Source code**
تعليمات برنامج مكتوبة بلغة مصدرية مثل البيسيك (BASIC) والكوبول (COBOL) والفورتران (FORTRAN) وباسكال Pascal.

خزن مؤقت سريع **Spooling**
في الدخل، نقل المعطيات الى خزن مساعد واحتجازها للمعالجة النهائية. وفي الخرج، نقل المعطيات الى خزن مساعد لاخراجها نهائياً الى جهاز خرج. أسلوب يستعمل لزيادة فعالية المعالجة الدفعية.

خزن **Storage**
ذاكرة.

برنامج مخزون **Stored program**
سلسلة من التعليمات موضوعة في ذاكرة الكمبيوتر الرئيسية للتحكم بذلك الكمبيوتر. يميز البرنامج المخزون الكمبيوتر عن الحاسب الآلي.

نظام **System**
مجموعة من المكونات التي تؤدي وظائفها مجتمعة لتحقيق هدف ما.

مخطط سير عمليات النظام **System flowchart**
تمثيل بياني لنظام مادي تمثل الرموز فيه برامج ومكونات كيان مادي وملفات الخ...

كيان منطقي نظامي **System software**
كيان منطقي مساند، على سبيل المثال، نظام التشغيل. قارن بـApplication program.

تحليل الأنظمة Systems analysis
عملية تحويل احتياجات المستخدم الى نظام يعمل.

محلل الأنظمة Systems analyst
مهني يترجم احتياجات المستخدم الى الخصائص الفنية المطلوبة لتطبيق نظام ما.

الاتصالات السلكية واللاسلكية Telecommunication
ارسال الاشارات عبر المسافات بواسطة خطوط التلفون أو اشارات الراديو أو غيرها من الوسائط.

طرفية Terminal
كيان مادي موضوع عند نقطة الدخول أو الخروج في شبكة اتصالات لغرض ادخال المعطيات أو الحصول عليها.

المشاركة الزمنية Time-sharing
سلسلة من الأساليب التي تسمح للمستخدمين المتعددين الذين يتحكم كل منهم بطرفية بأن يتشاركوا كمبيوترا واحدا. ان النقل الى الذاكرة والنقل من الذاكرة أسلوب شائع لادارة الذاكرة، كما أن توزيع الحصص الزمنية أسلوب شائع لادارة المعالج. وغالباً يتم التحكم في نيل الطرفيات بواسطة الاستطلاع.

توزيع الحصص الزمنية Time-slicing
أسلوب لادارة المعالج يعطى فيه برنامج تطبيقي «حصة» زمنية مستقلة ليكمل عمله فيها. واذا لم ينجز العمل أثناء حصة زمنية واحدة، فان البرنامج يفقد التحكم في المعالج ويجب أن يعود الى مؤخرة الصف لينتظر دوراً آخر.

رميز Token
اشارة الكترونية غالباً ما تكون جزءاً من بروتوكول. انظر Taken-passing network.

شبكة تمرير الرميز Token-passing network
شبكة اتصالات تقرر حق الارسال فيها اشارة الكترونية تسمى الرميز تمرر من طرفية الى طرفية. ان الطرفية التي تحمل الرميز هي وحدها القادرة على ارسال المعطيات على الخط.

سكة Track
واحدة من سلسلة دوائر متحدة المركز على قرص أو أسطوانة تخزن المعطيات حولها. وغالباً ما تقسم السكك الى قطاعات.

معاملات Transaction
تبادل بين مستخدم وكمبيوتر ينجز وظيفة منطقية واحدة. على سبيل المثال، إن كل الخطوات التي يشتمل عليها طلب النقود من آلة صرف نقود أوتوماتية تشكل معاملة واحدة.

معالجة المعاملات Transaction processing
معالجة المعاملات ما أن تحدث بدلاً من معالجتها على دفعات. قارن بـBatch processing.

ارسال المعطيات Transmission
بث المعطيات من موقع الى آخر على خط اتصالات.

وحدة التحكم بالارسال Transmission control unit or TCU
وحدة تحكم تصل خط أو خطوط الاتصال بنظام كمبيوتر.

مستخدم User
شخص أو مجموعة تستخدم برنامجاً أو نظاماً.

قرص بصري Video disk
وسط خزن مساعد يسجل المعطيات ويقرأها مستخدماً شعاع ليزر.

ذاكرة ظاهرية Virtual memory
أسلوب لادارة الذاكرة تحمل فيه الأجزاء النشطة فقط من البرنامج في الذاكرة الرئيسية.

دخل/خرج صوتي Voice I/O
دخل أو خرج الأصوات المحكية.

ذاكرة غير مستقرة Volatile memory
ذاكرة تفقد محتوياتها عندما تقطع الطاقة.

حالة انتظار Wait state
وضع يجب أن تنتظر بموجبه مهمة أو معالجة معينة اكتمال حدث قبل أن تستأنف.

قرص ونشستر Winchester disk
نوع من أنواع القرص المغنطيسي يجمع سطحه، أو أسطحه، مع مجموعته الخاصة من رؤوس القراءة/الكتابة، واستخدامه واسع الانتشار في أنظمة الميكروكمبيوتر.

كلمة Word
وحدة الخزن الأساسية التي يصمم نظام الكمبيوتر على أساسها، وتتكون الكلمة من خانتين أو أكثر في جميع الميكروكمبيوترات ما عدا الأصغر منها.

مفاهيم الكمبيوتر الأساسية

William S. Davis وليام س. ديفيز
جامعة ميامي
أكسفورد - ولاية أوهايو



الطبعة العربية الأولى
صادرة عن
مؤسسة الأبحاث اللغوية
Language Management Corporation
28 Chitron St., Tofarco House B, Suite 42, P.O.Box 7238, Nicosia Cyprus,
Telephone: (3572) 449828, Telecopier: (3572) 461751, Telex: 5860 LMC CY.
بالاتفاق مع شركة أديسون - ويسلي للنشر
Addison-Wesley Publishing Company

رقم التسجيل الدولي للكتاب ISBN 0-942517-01-6 New york.

تمهيد

لا حاجة الى تأكيد أهمية المكانة التي احتلّها الكمبيوتر في حياتنا اليومية، حيث أصبح «أداة العصر الحديث»، وقد اشتدّ اقبال الناس على تعلم دقائق هذا الجهاز وكيفية الافادة منه. ويتّجه الكثيرون أولاً الى الاحاطة بالمفاهيم الأساسية.. والواقع أن المهندسين، ورجال الأعمال، والعلماء، والعاملين في حقل الرياضيات، والدارسين والطلبة، يعتبرون الكمبيوتر كأداة هامّة في مهنهم، وهذه الأداة تحتاج الى الكثير من التدرّب عليها. أمّا الذين اتخذوا من الكمبيوتر مهنة لهم، وكذلك الدارسون المتضلّعون في حقول البحث الكمبيوتري، فان عليهم، طبعاً، أن يتعمّقوا في دراساتهم. ومن الواضح أن برنامجاً مُعدًّا لطلبة ادارة الأعمال يختلف عن برنامج معدّ للمهندسين، أو لعلماء الكمبيوتر، أو لدارسي الفنون الحرّة. ولهذا كان على مدارسنا أن تقدّم مداخل مختلفة لدروس الكمبيوتر.

غير أنه رغم اختلاف هذه المناهج فانها تتفق على مجموعة مشتركة من المفاهيم الأساسية. وهذه المفاهيم الأساسية تكفي لكثيرين من الطلبة. فان طلبة العلوم والهندسة وادارة الأعمال والرياضيات، يعتمدون عليها لادراك تطبيقات الكمبيوتر كل في حقله. وحتى بالنسبة الى طلبة النظم الكمبيوترية، فان هذه المفاهيم الأساسية نفسها تتيح أساساً متيناً للدراسات المستقبلية، إذ ان المفاهيم الأساسية الراسخة هي نفسها لا تتغير بتغيّر المنهج.

وهناك برامج مبدئية متعدّدة تعزّز هذه المناهج الأساسية بموضوعات أخرى، فبعضها يؤكّد على المضمون الاجتماعي، وبعضها الآخر يؤكّد على لغة البرمجة أو يطلع الطلبة على تطبيقات الكمبيوتر في حقل معين. وهناك كتب جيّدة عديدة قليلة التكلفة، تبحث في لغات البرمجة والتطبيقات الأخرى. ويمكن استخدام هذه الكتب مع كتاب يتضمّن مفاهيم الكمبيوتر الأساسية بحيث يمكن دعم أي منهج مبدئي للكمبيوتر.

وهذه هي الفكرة الأساسية التي ينطوي عليها هذا الكتاب «مفاهيم الكمبيوتر الأساسية». فقد أعد هذا الكتاب بحيث يكون مدخلاً جيداً قليل التكلفة لعالم الكمبيوترات، دقيق العبارة، شاملاً من الناحية الفنية وموجزاً.

ويسرّ «مؤسسة الأبحاث اللغوية» أن تقدّم هذا الكتاب بالعربية الى جميع الطلبة العرب والباحثين في علم الكمبيوتر في هذا العصر الالكتروني.

المحتويات